

|              |              |         |
|--------------|--------------|---------|
| SSSSSSSSSSSS | DDDDDDDDDDDD | AAAAAAA |
| SSSSSSSSSSSS | DDDDDDDDDDDD | AAAAAAA |
| SSSSSSSSSSSS | DDDDDDDDDDDD | AAAAAAA |
| SSS          | DDD          | AAA     |
| SSSSSSSSSS   | DDD          | AAA     |
| SSSSSSSSSS   | DDD          | AAA     |
| SSSSSSSSSS   | DDD          | AAA     |
| SSS          | DDD          | AAA     |
| SSSSSSSSSSSS | DDDDDDDDDDDD | AAAAAAA |
| SSSSSSSSSSSS | DDDDDDDDDDDD | AAAAAAA |
| SSSSSSSSSSSS | DDDDDDDDDDDD | AAAAAAA |

\*\*FILE\*\*ID\*\*LOCK

M 4

LL 000000 000000 CCCCCCCC CCCC KK KK  
LL 000000 000000 CCCCCCCC CCCC KK KK  
LL 00 00 CC KK KK  
LL 000000 000000 CCCCCCCC CCCC KK KK  
LL 000000 000000 CCCCCCCC CCCC KK KK

.....  
.....  
.....

LL IIIII SSSSSSSS SSSSSSSS  
LL IIIII SS SSSSSS SSSSSS  
LL IIIII SS SSSSSS SSSSSS

LOCK  
VO4-  
65 E  
E  
65 E

|      |      |  |
|------|------|--|
| (1)  | 2    | COPYRIGHT NOTICE                                       |
| (1)  | 29   | PROGRAM DESCRIPTION                                    |
| (2)  | 77   | DECLARATIONS   |
| (3)  | 103  | STORAGE DEFINITIONS                                    |
| (5)  | 342  | READ-ONLY DATA DEFINITIONS                             |
| (6)  | 465  | SHOW_ALL_LOCKS - ACTION ROUTINE TO DISPLAY ALL LOCKS   |
| (7)  | 499  | SHOW_ONE_LOCK - ACTION ROUTINE TO DISPLAY ONE LOCK     |
| (8)  | 529  | SHOW_PROC_LOCK - SHOW LOCKS ASSOCIATED WITH A PROCESS  |
| (9)  | 572  | GET_LKB - GET LOCK BLOCK                               |
| (10) | 618  | SAVE_LOCK_DATA - RETRIEVE LOCK DATA FROM SYSTEM        |
| (11) | 658  | DISPLAY_LOCK - CONTROL DISPLAY OF LOCK DATA            |
| (12) | 683  | FORMAT_LOCK - FORMAT LOCK DATA                         |
| (13) | 867  | PRINT_LOCK - OUTPUT LOCK DATA                          |
| (14) | 990  | LOCK_COND HAND - CONDITION HANDLER FOR SHOW_ALL_LOCKS  |
| (15) | 1029 | SHOW_ALL_RES - ACTION ROUTINE TO DISPLAY ALL RESOURCES |
| (16) | 1091 | SHOW_ONE_RES - ACTION ROUTINE TO DISPLAY ONE RESOURCE  |
| (17) | 1124 | SHOW_RSB - CONTROL FOR RSB DISPLAY                     |
| (18) | 1147 | SHOW_QUEUES - DISPLAY QUEUES FOR GIVEN RESOURCE        |
| (19) | 1180 | FORMAT_RSB - FORMAT RSB DATA                           |
| (20) | 1257 | PRINT_RSB - OUTPUT RSB DATA                            |
| (21) | 1358 | PROCESS_QUEUE - TRAVERSE RESOURCE QUEUES               |
| (22) | 1428 | PRINT_LINE - OUTPUT QUEUE DATA                         |

0000 1 .TITLE LOCK LOCK AND RESOURCE FORMATTING ROUTINES  
0000 2 .SBTTL COPYRIGHT NOTICE  
0000 3 .IDENT 'V04-00'  
0000 4 :  
0000 5 :\*\*\*\*\*  
0000 6 :\*  
0000 7 :\* COPYRIGHT (c) 1978, 1980, 1982, 1984 BY  
0000 8 :\* DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.  
0000 9 :\* ALL RIGHTS RESERVED.  
0000 10 :\*  
0000 11 :\* THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  
0000 12 :\* ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE  
0000 13 :\* INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER  
0000 14 :\* COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  
0000 15 :\* OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY  
0000 16 :\* TRANSFERRED.  
0000 17 :\*  
0000 18 :\* THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE  
0000 19 :\* AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT  
0000 20 :\* CORPORATION.  
0000 21 :\*  
0000 22 :\* DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS  
0000 23 :\* SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.  
0000 24 :\*  
0000 25 :\*  
0000 26 :\*\*\*\*\*  
0000 27 :\*

0000 29 .SBTTL PROGRAM DESCRIPTION  
0000 30 ::++  
0000 31 : Facility  
0000 32 : System Dump Analyzer  
0000 33 :  
0000 34 : Abstract  
0000 35 :  
0000 36 : Routines to Format Lock and Resource Information  
0000 37 :  
0000 38 : Environment  
0000 39 :  
0000 40 : Native Mode, User Mode  
0000 41 :  
0000 42 : Author  
0000 43 :  
0000 44 : Maryann Hinden, June 1982  
0000 45 :  
0000 46 :  
0000 47 : Modified By  
0000 48 :  
0000 49 : V03-007 SRB0109 Steve Beckhardt 2-Feb-1984  
0000 50 : Fixed formatting of Directory entry text in SHO RES Line.  
0000 51 :  
0000 52 : V03-006 SRB0107 Steve Beckhardt 7-Dec-1983  
0000 53 : Enhanced SHOW LOCK and SHOW RESOURCE to display  
0000 54 : distributed lock information.  
0000 55 :  
0000 56 : V03-005 SRB0089 Steve Beckhardt 27-May-1983  
0000 57 : Fixed bug displaying resource name. Added new status bit  
0000 58 : definitions.  
0000 59 :  
0000 60 : V03-004 SRB0071 Steve Beckhardt 16-Mar-1983  
0000 61 : Changed the way system resources are decoded in RSB  
0000 62 : to use zero group number instead of SYSNAM flag.  
0000 63 :  
0000 64 : V03-003 MSH0003 Maryann Hinden 17-Dec-1982  
0000 65 : Fix broken assume's.  
0000 66 :  
0000 67 : V03-002 MSH0002 Maryann Hinden 22-Oct-1982  
0000 68 : Determine empty lock queue correctly in SHOW\_PROC\_LOCK.  
0000 69 : Change lock display - output lock and parent-id in hex;  
0000 70 : rearrange format.  
0000 71 :  
0000 72 : V03-001 MSH0001 Maryann Hinden 16-Jul-1982  
0000 73 : Correct references to LCK\$xxxx data structures; change  
0000 74 : condition handling.  
0000 75 :--

0000 77 .SBTTL DECLARATIONS  
0000 78 :  
0000 79 : SYSTEM SYMBOL DEFINTIONS  
0000 80 :  
0000 81 :  
0000 82 \$CHFDEF  
0000 83 \$LCKDEF  
0000 84 \$LKBDEF  
0000 85 \$OPTDEF  
0000 86 \$PCBDEF  
0000 87 \$RSBDEF  
0000 88 \$STSDEF  
0000 89 :  
0000 90 :  
0000 91 : LOCAL SYMBOL DEFINITIONS  
0000 92 :  
0000 93 :  
FFFFFFFFFF 0000 94 GRANT = -1 : SYMBOLIC DESIGNATIONS FOR QUEUE TYPES  
00000000 0000 95 CONVERT = 0  
00000001 0000 96 WAIT = 1  
00000000 0000 97 :  
00000005 0000 98 LKID\_OFF = LKB\$L\_LKID-LKB\$L\_LKID : OFFSETS INTO SAVED QUEUE DATA  
00000004 0000 99 GRMD\_OFF = LKB\$B\_GRMODE-LKB\$L\_LKID  
00000000 0000 100 RQMD\_OFF = LKB\$B\_RQMODE-LKB\$L\_LKID  
00000000 0000 101 :

63 6F 6C 20 66 6F 0000009C'010E0000'  
 73 79 73 20 6E 6F 20 4C 58 21 20 6B  
 4C 58 21 20 6D 65 74

```

00000000 103 .SBTTL STORAGE DEFINITIONS
00000000 104 .PSECT SDADATA,NOEXE,WRT,LONG
00000000 105 :
00000000 106 : Data area for SHOW LOCKS commands
00000000 107 :
00000000 108 :
00000000 109 LOCK_COUNT: ; count of locks processed
00000000 110 .LONG 0
00000000 111 PARID_BFR: ; parent lock id
00000000 112 .LONG 0
00000000 113 LKB_BFR: ; lock block data
00000068 114 .BLKB LKB$K_LENGTH
00000068 115 :
0000006C 116 RSBCSID_BFR: ; CSID in RSB
0000006C 117 .BLKL 1
0000006C 118 LKB_RSB_BFR: ; resource block data for given lock
0000006C 119 GROUP_BFR: group number
0000006E 120 .BLKW 1
0000006F 121 RMOD_BFR: ; access mode
0000006F 122 .BLKB 1
00000070 123 RSNLEN_BFR: ; resource name length
00000070 124 .BLKB 1
00000070 125 RESNAM_BFR: ; resource name
00000090 126 .BLKB RSB$K_MAXLEN+1
00000090 127 :
00000094 128 LOCKID::; lock ID currently being processed
00000094 129 .BLKL 1
00000094 130 :
00000094 131 .ALIGN LONG
00000094 132 :
00000094 133 FAO_RMINFO_DSC: ; ASCII "of lock !XL on system !XL"
00000094 134 .ASCID "of lock !XL on system !XL"

00000000 135 :
00000000 136 .ALIGN LONG
00000000 137 RMINFO_LEN:
00000000 138 .LONG 0
00000050 139 RMINFO_DSC:
00000050 140 .LONG 80
000000C5 141 .ADDRESS RMINFO_BFR
000000C5 142 :
000000C5 143 RMINFO_CNT:
000000C5 144 .BLKB 1
000000C5 145 RMINFO_BFR:
000000C5 146 .BLKB 80
00000115 147 :
00000115 148 :
00000115 149 : Data area for SHOW RESOURCES commands
00000115 150 :
00000115 151 :
00000000 152 RES_COUNT: ; count of resources processed
00000000 153 .LONG 0
00000000 154 QUEUE_COUNT: ; count of queue elements processed
00000000 155 .LONG 0
0000018C 156 RSB_BFR: ; resource block data
0000018C 157 .BLKB RSB$K_MAXLEN+RSB$K_LENGTH

```

```

00000190 018C 158 HTBL_INDX: ; current index into hash table
00000190 018C 159 .BLKL 1
00000194 0190 160 HTBL_CNT: ; size of hash table
00000194 0190 161 .BLKL 1
00000198 0194 162 COUNT: ; # queue items to display
00000198 0198 163 .BLKL 1
0000019C 0198 164 QUEUE_TYPE: ; type of queue being processed
0000019C 019C 165 .BLKL 1
0000019F 019C 166 GRMD_BFR: ; grant mode
0000019F 019F 167 RQMD_BFR: ; requested mode
000001A2 019F 168 .BLKB 3
000001A2 01A2 169 .BLKB 3
000001AE 01A2 170 LKID_BFR: ; lock id
000001AE 01AE 171 .BLKL 3
000001BD 01AE 172 FAO_GROUP_DSC: ; FAO data storage for GROUP display
000001BD 01BA 173 .ASCID "'!30W"
000001BD 01BA 174
000001BD 01BA 175 GROUP_BUF: ; FAO data storage for RESOURCE display
000001BD 01BA 176 .BLKB 3
00000003 01BD 177
00000003 01CO 178 .ALIGN LONG
000001BA 01CO 179 GROUP_BUF_DSC: ; FAO data storage for RESOURCE display
000001BA 01C4 180 .LONG 3
000001BA 01C8 181 .ADDRESS GROUP_BUF
000001BA 01C8 182
000001BA 01C8 183 SYS_DSC: ; FAO data storage for RESOURCE display
000001BA 01C8 184 .ASCID "System"
000001BA 01D6 185
000001BA 01D6 186 .ALIGN LONG
0000000B 01D8 187 GROUP_DSC: ; FAO data storage for RESOURCE display
000001E0 01D8 188 .LONG 11
000001E0 01DC 189 .ADDRESS GROUP_TXT
000001E0 01E0 190
000001E0 01E0 191 GROUP_TXT: ; FAO data storage for RESOURCE display
000001E0 01E0 192 .ASCII "Group "
000001EB 01E8 193 GROUP_NUM: ; FAO data storage for RESOURCE display
000001EB 01E8 194 .BLKB 3
000001EB 01EB 195
000001EB 01EB 196 ; FAO data storage for RESOURCE display
000001EB 01EB 197 : FAO data storage for RESOURCE display
000001EB 01EB 198 :
000001EB 01EB 199
000001EB 01EB 200 : Line 1
000001EB 01EB 201
00000000 01EB 202 RSB: .LONG 0 ; Address of RSB
00000000 01EF 203 GGMOD: .LONG 0 ; Group grant mode (addr of ASCII string)
00000000 01F3 204
00000000 01F3 205 : Line 2
00000000 01F3 206
00000000 01F3 207 PRSB: .LONG 0 ; Address of parent RSB
00000000 01F7 208 CGMOD: .LONG 0 ; Conversion grant mode (addr of ASCII string)
00000000 01FB 209
00000000 01FB 210 : Line 3
00000000 01FB 211
00000000 01FB 212 SRSBCT: .LONG 0 ; Sub-RSB reference count
00000000 01FF 213 BAST: .LONG 0 ; Blocking AST count
00000000 0203 214

```

|           |      |     |                  |  |
|-----------|------|-----|------------------|--|
| 0203      | 215  | :   | Line 4           |  |
| 000000000 | 0204 | 216 |                  |  |
| 000000000 | 0207 | 217 | VAL4: .LONG 0    | : Value block longword #4                      |
| 000000000 | 020B | 218 | VAL3: .LONG 0    | : Value block longword #3                      |
| 000000000 | 020F | 219 | VAL2: .LONG 0    | : Value block longword #2                      |
| 000000000 | 0213 | 220 | VAL1: .LONG 0    | : Value block longword #1                      |
| 000000000 | 0217 | 221 | SEQNUM: .LONG 0  | : Sequence number                              |
| 000000000 | 0218 | 222 | VALID: .LONG 0   | : Value block valid (addr. of .ASCII string)   |
|           | 021B | 223 |                  |  |
|           | 021B | 224 | :                | Line 5   |
| 000000000 | 021B | 225 |                  |  |
| 000000000 | 021F | 226 | RESN2: .LONG 0   | : Second longword of resource name             |
| 000000008 | 0223 | 227 | RESN1: .LONG 0   | : First  |
| 000000000 | 0227 | 228 |                  | : Count of ASCII string                        |
| 000000000 | 0228 | 229 | BUF1: .LONG 0    | : Text of resource name (addr of ASCII string) |
|           | 022B | 230 |                  |  |
|           | 022B | 231 | :                | Line 6   |
| 000000000 | 022B | 232 |                  |  |
| 000000000 | 022F | 233 | RESNLEN:.LONG 0  | : Resource name length                         |
| 000000000 | 0233 | 234 | RESN4: .LONG 0   | : Fourth longword of resource name             |
| 000000000 | 0233 | 235 | RESN3: .LONG 0   | : Third  |
| 000000008 | 0237 | 236 |                  | : Count of ASCII string                        |
| 000000000 | 0238 | 237 | BUF2: .LONG 0    | : Text of resource name (addr of ASCII string) |
| 000000000 | 023F | 238 | CSID: .LONG 0    | : RSB CSID                                     |
|           | 0243 | 239 |                  |  |
|           | 0243 | 240 | :                | Line 7   |
|           | 0243 | 241 |                  |  |
| 000000000 | 0243 | 242 | RACMOD: .LONG 0  | : Resource access mode (addr of ASCII string)  |
| 000000000 | 0247 | 243 | RESN6: .LONG 0   | : Sixth longword of resource name              |
| 000000000 | 024B | 244 | RESN5: .LONG 0   | : Fifth  |
| 000000008 | 024F | 245 |                  | : Count of ASCII string                        |
| 000000000 | 0253 | 246 | BUF3: .LONG 0    | : Text of resource name (addr of ASCII string) |
| 000000000 | 0257 | 247 | DIRENTRY:        |  |
| 000000000 | 0257 | 248 |                  | : Directory entry (addr. of .ASCII string)     |
|           | 025B | 249 |                  |  |
|           | 025B | 250 | :                | Line 8   |
|           | 025B | 251 |                  |  |
| 000000000 | 025B | 252 | RNSPACE: .LONG 0 | : Address of descriptor                        |
| 000000000 | 025F | 253 | RESN8: .LONG 0   | : Eighth longword of resource name             |
| 000000000 | 0263 | 254 | RESN7: .LONG 0   | : Seventh                                      |
| 000000008 | 0267 | 255 |                  | : Count of ASCII string                        |
| 000000000 | 026B | 256 | BUF4: .LONG 0    | : Text of resource name (addr of ASCII string) |
|           | 026F | 257 |                  |  |

```

026F 259 :
026F 260 : FAO data storage for LOCKS display
026F 261 :
026F 262 :
026F 263 : LINE 1
026F 264 :
00000000 026F 265 LKID: .LONG 0 : Lock id (value)
00000000 0273 266 PID: .LONG 0 : PID (value)
00000000 0277 267 FLAGS1: .LONG 0 : First line of flags (addr. of .ASCII string)
00000000 027B 268 .LONG 0 :
00000000 027F 269 .LONG 0 : " " " "
0283 270 :
0283 271 : LINE 2
0283 272 :
00000000 0283 273 PARID: .LONG 0 : Parent id (value)
00000000 0287 274 STATE1: .LONG 0 : Lock state info (address of .ASCII string)
00000002 028B 275 .LONG 2 : Lock mode (length of string)
00000000 028F 276 MODE1: .LONG 0 : Lock mode (address of .ASCII string)
00000000 0293 277 FLAGS2: .LONG 0 : Second line of flags (addr. of .ASCII string)
00000000 0297 278 .LONG 0 :
00000000 0298 279 .LONG 0 : " " " "
029F 280 :
029F 281 : LINE 3
029F 282 :
00000000 029F 283 SUBLKS: .LONG 0 : Sublocks (value)
00000000 02A3 284 STATE2: .LONG 0 : More lock state info (addr. of .ASCII string)
00000002 02A7 285 .LONG 2 : More lock mode (length of string)
00000000 02AB 286 MODE2: .LONG 0 : More lock mode (addr. of .ASCII string)
00000000 02AF 287 FLAGS3: .LONG 0 : Third line of flags (addr. of .ASCII string)
00000000 02B3 288 .LONG 0 :
00000000 02B7 289 .LONG 0 : " " " "
02BB 290 :
02BB 291 : LINE 3A
02BB 292 :
00000000 02BB 293 BLKAST: .LONG 0 : Blocking AST (address of .ASCII string)
00000000 02BF 294 LKB: .LONG 0 : Address of LKB
02C3 295 :
02C3 296 : LINE 4
02C3 297 :
00000000 02C3 298 RES2: .LONG 0 : Second longword of resource name (value)
00000000 02C7 299 RES1: .LONG 0 : First longword of resource name (value)
00000008 02CB 300 .LONG 8 :
00000000 02CF 301 DMP1: .LONG 0 : First line of text name (.ASCII string)
00000000 02D3 302 STS1: .LONG 0 : 1st line of status (addr. of .ASCII strings)
00000000 02D7 303 .LONG 0 :
00000000 02DB 304 .LONG 0 :
02DF 305 :
02DF 306 : LINE 5
02DF 307 :
00000000 02DF 308 RLEN: .LONG 0 : Resource name length (value)
00000000 02E3 309 RES4: .LONG 0 : Fourth longword of resource name (value)
00000000 02E7 310 RES3: .LONG 0 : Third longword of resource name (value)
00000008 02EB 311 .LONG 8 :
00000000 02EF 312 DMP3: .LONG 0 : Second line of text name (.ASCII string)
00000000 02F3 313 STS2: .LONG 0 : 2nd line of status (addr. of .ASCII strings)
00000000 02F7 314 .LONG 0 :
00000000 02FB 315 .LONG 0 :

```

|          |     |               |   |
|----------|-----|---------------|---|
| 02FF     | 316 |               |   |
| 02FF     | 317 | ; LINE 6      |   |
| 00000000 | 319 | ACMODE: .LONG | 0 |
| 00000000 | 320 | RES6: .LONG   | 0 |
| 00000000 | 321 | RESS: .LONG   | 0 |
| 00000008 | 322 | .LONG         | 8 |
| 00000000 | 323 | DMP5: .LONG   | 0 |
| 00000000 | 324 | STS3: .LONG   | 0 |
| 00000000 | 325 | .LONG         | 0 |
| 00000000 | 326 | .LONG         | 0 |
| 00000000 | 327 |               |   |
| 031F     | 328 | ; LINE 7      |   |
| 031F     | 329 |               |   |
| 00000000 | 330 | NSPACE: .LONG | 0 |
| 00000000 | 331 | RES8: .LONG   | 0 |
| 00000000 | 332 | RES7: .LONG   | 0 |
| 00000008 | 333 | .LONG         | 8 |
| 00000000 | 334 | DMP7: .LONG   | 0 |
| 0333     | 335 |               |   |
| 0333     | 336 | ; LINE 8      |   |
| 0333     | 337 |               |   |
| 00000000 | 338 | TYPE: .LONG   | 0 |
| 00000000 | 339 | RMINFO: .LONG | 0 |
| 033B     | 340 |               |   |

; Access mode (Address of .ASCII string)  
; Sixth longword of resource name (value)  
; Fifth longword of resource name (value)  
; Third line of text name (.ASCII string)  
; 3rd line of status (addr. of .ASCII strings)  
  
; Name space (group/system) (addr. of desc.)  
; Eighth longword of resource name (value)  
; Seventh longword of resource name (value)  
; Fourth line of text name (.ASCII string)  
  
; Lock type (addr. of .ASCII string)  
; Remote info (addr. of .ASCII string)

```

033B 342 .SBttl READ-ONLY DATA DEFINITIONS
00000000 343 .Psect LOCK,EXE,NOWRT,LONG
0000 344
0000 345 : HEADERS
0000 346
0015 347 LOCK_HEAD: STRING <Lock database>
002E 348 RES_READ: STRING <Resource database>
002E 349
00000000 350 LOCK_STR_TBL: ; table of FAO strings for wait/grant queue display
002F 351 .LONG 0
0000005A' 352 .LONG LOCKSTR1
0032 353 .LONG LOCKSTR2
0000006F' 354 .LONG LOCKSTR3
0036
00000093' 355
003A
003E
00000000 356 CONV_STR_TBL: ; table of FAO strings for convert queue display
003E 357 .LONG 0
000000C6' 358 .LONG CONVSTR1
0042 359 .LONG CONVSTR2
000000DF' 360 .LONG CONVSTR3
0046
00000108' 361
004A
004E
00000141' 362 QUE_STR_TBL: ; table of headers for queue display
004E 363 .LONG GRANTSTR
0000016B' 364 .LONG CONVSTR
0052 365 .LONG WAITSTR
0056
005A
366
21 20 20 20 20 20 00000062'010E0000' 367 LOCKSTR1: .ASCID # !XL !AC#
43 41 21 20 20 4C 58 0068
21 20 20 20 20 00000077'010E0000' 368 LOCKSTR2: .ASCID # !XL !AC !XL !AC#
20 20 20 43 41 21 20 20 4C 58 007D
43 41 21 20 20 4C 58 21 20 20 0089
21 20 20 20 20 00000098'010E0000' 369 LOCKSTR3: .ASCID # !XL !AC !XL !AC !XL !AC#
20 20 20 43 41 21 20 20 4C 58 0093
20 20 43 41 21 20 20 4C 58 00A1
20 20 43 41 21 20 20 4C 58 00AD
41 21 20 20 4C 58 21 20 20 20 20 0089
43 00C5
00C6
21 20 20 20 20 000000CE'010E0000' 370 CONVSTR1: .ASCID # !XL !AC!/AC#
43 41 21 2F 43 41 21 20 20 4C 58 00D4
21 20 20 20 20 000000E7'010E0000' 371 CONVSTR2: .ASCID # !XL !AC!/AC !XL !AC!/AC#
20 43 41 21 2F 43 41 21 20 20 4C 58 00ED
2F 43 41 21 20 20 4C 58 21 20 20 00F9
43 41 21 20 20 4C 58 21 20 20 0105
21 20 20 20 20 00000110'010E0000' 372 CONVSTR3: .ASCID # !XL !AC!/AC !XL !AC!/AC !XL !AC!/AC#
20 43 41 21 2F 43 41 21 20 20 4C 58 0108
2F 43 41 21 20 20 4C 58 21 20 20 0116
20 20 4C 58 21 20 20 20 43 41 21 0122
43 41 21 2F 43 41 21 012E
013A
0141
65 74 6E 61 72 47 00000149'010E0000' 373 GRANTSTR: .ASCID #Granted queue (Lock ID / Gr mode):#
63 6F 4C 28 20 65 75 65 75 71 20 64 014F
6F 6D 20 72 47 20 2F 20 44 49 20 68 015B
3A 29 65 64 0167
72 65 76 6F 43 00000173'010E0000' 0168
28 20 65 75 75 71 20 6E 6F 69 73 0179
72 47 20 2F 20 44 49 20 68 63 6F 4C 0185
3A 29 65 64 6F 6D 20 71 52 2F 0191
6E 69 74 69 61 57 000001A3'010E0000' 0198
374
375 WAITSTR: .ASCID #Waiting queue (Lock ID / Rq mode):#
376 CONVSTR: .ASCID #Conversion queue (Lock ID / Gr/Rq mode):#
377

```

```

63 6F 4C 28 20 65 75 65 75 71 20 67 01A9
6F 6D 20 71 52 20 2F 20 44 49 20 6B 01B5
3A 29 65 64 01C1
01C5 378
01C5 379
01C5 380
01C5 381
01C5 382 Generic Data
4C 4E 01C5 383 .ASCII "NL"
52 43 01C7 384 .ASCII "CR"
57 43 01C9 385 .ASCII "CW"
52 50 01CB 386 .ASCII "PR"
57 50 01CD 387 .ASCII "PW"
58 45 01CF 388 .ASCII "EX"
01D1 389
000001E9 01D1 390 LKMODE_TBL: .ADDRESS NULL ; POINTERS TO MODE TEXT
000001EC 01D5 391 .ADDRESS CREAD
000001EF 01D9 392 .ADDRESS CWRITE
000001F2 01DD 393 .ADDRESS PREAD
000001F5 01E1 394 .ADDRESS PWRITE
000001F8 01E5 395 .ADDRESS EX
4C 4E 00 01E9 396 NULL: .ASCIC /NL/
02 01E9
52 43 00 01EC 397 CREAD: .ASCIC /CR/
02 01EC
57 43 00 01EF 398 CWRITE: .ASCIC /CW/
02 01EF
52 50 00 01F2 399 PREAD: .ASCIC /PR/
02 01F2
57 50 00 01F5 400 PWRITE: .ASCIC /PW/
02 01F5
58 45 00 01F8 401 EX: .ASCIC /EX/
02 01F8
01FB 402
01FB 403 GR_STRING: .ASCIC "Granted at"
0A 01FB 404 .ASCIC "Granted at"
0206 405 CV_STRING: .ASCIC "Converting to"
6F 74 0206 406 .ASCIC "Converting to"
0D 0212
0206 407 WT_STRING: .ASCIC "Waiting for"
0214 408 .ASCIC "Waiting for"
0B 0214
0214 409
20 20 0220 410 BLANKS: .ASCII " "
00 0222 411
00 0222 412 NULL_CSTRING: .ASCIC ...
00 0223 413 .ASCIC ...
0223 414
0223 415 BL_STRING: .ASCIC "BLKAST"
06 0223 416 .ASCIC "BLKAST"
022A 417
022A 418 FLAGS_TBL: .ASCIC "VALBLK"
022A 419 .ASCIC "VALBLK"

```

|                      |         |                              |
|----------------------|---------|------------------------------|
| 54 52 45 56 4E 4F 43 | 07 022A | 420 .ASCIC "CONVERT"         |
| 45 55 45 55 51 4F 4E | 00 0232 | 421 .ASCIC "NOQUEUE"         |
| 53 54 53 43 4E 59 53 | 00 023A | 422 .ASCIC "SYNCSTS"         |
| 20 4D 45 54 53 59 53 | 00 0242 | 423 .ASCIC "SYSTEM "         |
| 41 54 4F 55 51 4F 4E | 07 024A | 424 .ASCIC "NOQUOTA"         |
| 20 53 59 53 54 56 43 | 07 0252 | 425 .ASCIC "CVTSYS "         |
| 52 45 56 4F 43 45 52 | 00 025A | 426 .ASCIC "RECOVER"         |
| 54 43 45 54 4F 52 50 | 07 0262 | 427 .ASCIC "PROTECT"         |
|                      | 07 026A | 428                          |
|                      | 0272    | 429 STATUS_TBL:              |
| 54 53 41 4C 50 43 44 | 00 0272 | 430 .ASCIC "DCPLAST"         |
| 54 53 41 4B 4C 42 44 | 07 0272 | 431 .ASCIC "DBLKAST"         |
| 20 20 43 4E 59 53 41 | 00 027A | 432 .ASCIC "ASYNC "          |
| 44 51 54 53 41 4C 42 | 07 028A | 433 .ASCIC "BLASTQD"         |
| 20 59 50 43 54 53 4D | 00 0292 | 434 .ASCIC "MSTCPY "         |
| 41 54 4F 55 51 4F 4E | 07 029A | 435 .ASCIC "NOQUOTA"         |
| 51 54 55 4F 4D 49 54 | 00 02A2 | 436 .ASCIC "TIMOUTQ"         |
| 20 53 59 53 53 41 57 | 07 02AA | 437 .ASCIC "WASSYS "         |
| 53 59 53 4F 54 56 43 | 00 02B2 | 438 .ASCIC "CVTOSYS"         |
|                      | 07 02B2 | 439                          |
| 6C 61 63 6F 4C       | 00 02BA | 440 LOCAL: .ASCIC "Local"    |
|                      | 05 02BA |                              |
| 73 73 65 63 6F 72 50 | 00 02C0 | 441 PROCESS:                 |
|                      | 07 02C0 | 442 .ASCIC "Process"         |
| 72 65 74 73 61 4D    | 00 02CB | 443 MASTER: .ASCIC "Master"  |
|                      | 06 02CB |                              |
|                      | 02CF    | 444                          |
| 20 00                | 02CF    | 445 SPACE: .ASCIC " "        |
|                      | 01 02CF |                              |
|                      | 02D1    | 446                          |
| 64 69 6C 61 76 20 74 | 00 02D1 | 447 NOT_VALID:               |
| 6F 4E                | 09 02D1 | 448 .ASCIC "Not valid"       |
| 65 20 79 72 6F 74 63 | 00 02DB | 449 DIR_ENTRY:               |
| 65 72 69 44          | 79 02DB | 450 .ASCIC "Directory entry" |
|                      | 72 02E7 |                              |
|                      | OF 02DB |                              |
|                      | 02EB    | 451                          |

LOCK  
V04-000

LOCK AND RESOURCE FORMATTING ROUTINES  
READ-ONLY DATA DEFINITIONS

M S

16-SEP-1984 01:31:38 VAX/VMS Macro V04-00  
5-SEP-1984 03:32:46 [SDA.SRC]LOCK.MAR;1

Page 12  
(5)

|                                      |      |                                 |
|--------------------------------------|------|---------------------------------|
| 65 64 6F 6D 20 6C 65 6E 72 65 4B 00' | 02EB | 452 KMODE: .ASCIC "Kernel mode" |
|                                      | 0B   |                                 |
| 65 64 6F 6D 20 2E 63 65 78 45 00'    | 02EB | 453 EMODE: .ASCIC "Exec. mode"  |
|                                      | 0A   |                                 |
| 65 64 6F 6D 20 2E 72 65 70 75 53 00' | 02F7 | 454 SMODE: .ASCIC "Super. mode" |
|                                      | 0A   |                                 |
| 65 64 6F 6D 20 72 65 73 55 00'       | 0302 | 455 UMODE: .ASCIC "User mode"   |
|                                      | 0B   |                                 |
|                                      | 09   |                                 |
|                                      | 030E |                                 |
|                                      | 0318 | 456                             |
|                                      | 0318 | 457 ALIGN LONG                  |
|                                      | 0318 | 458 ACMODE_TBL:                 |
| 000002EB'                            | 0318 | 459 .ADDRESS KMODE              |
| 000002F7'                            | 031C | 460 .ADDRESS EMODE              |
| 00000302'                            | 0320 | 461 .ADDRESS SMODE              |
| 0000030E'                            | 0324 | 462 .ADDRESS UMODE              |
|                                      | 0328 | 463                             |

LOC  
V04

```

0328 465 .SBTTL SHOW_ALL_LOCKS - ACTION ROUTINE TO DISPLAY ALL LOCKS
0328 466 :+ ++
0328 467 :
0328 468 :
0328 469 :
0328 470 :
0328 471 : PURPOSE ACTION ROUTINE FOR THE "SHOW LOCKS/ALL" COMMAND
0328 472 :
0328 473 :
0328 474 : INPUT NONE
0328 475 :
0328 476 : OUTPUT DISPLAYED DATA
0328 477 :
0328 478 :
0328 479 :---
0000 0328 480 .ENTRY SHOW_ALL_LOCKS,"M<>
032A 481 :
6D 000009E8'EF 9E 032A 482 MOVAB LOCK_COND_HAND,(FP) : ESTABLISH CONDITION HANDLER
00000090'EF D4 0331 483 CLRL LOCKID : INITIALIZE VARIABLE
FCC5 CF 9F 0337 484 PUSHAB LOCK_HEAD : GET HEADER
00000000'EF 01 FB 033B 485 CALLS #1,SET HEADING : SET IT UP
00000000'EF D4 0342 486 CLRL LOCK_COUNT : INIT COUNT OF LOCKS PROCESSED
0348 487 :
0348 488 :
0348 489 : STEP THROUGH LOCK ID TABLE, DISPLAYING ENTRIES WITH DATA
0348 490 :
00000090'EF D6 0348 491 10$: INCL LOCKID : INCREMENT INDEX
00000443'EF 00 FB 034E 492 CALLS #0,GET_LKB : GET ADDRESS OF ASSOCIATED LKB
FO 50 E9 0355 493 BLBC R0,10$ : IF LBC NON-EXISTENT, TRY NEXT
00000000'EF D6 0358 494 INCL LOCK_COUNT : GOT ONE
000004B7'EF 00 FB 035E 495 CALLS #0,SAVE_LOCK_DATA : GET ASSOCIATED DATA
0000051D'EF 00 FB 0365 496 CALLS #0,DISPLAY_LOCK : DISPLAY DATA FOR THIS LOCKID
DA 11 036C 497 BRB 10$ : LOOP (SIGNAL WILL EXIT FOR US)

```

036E 499 .SBTTL SHOW\_ONE\_LOCK - ACTION ROUTINE TO DISPLAY ONE LOCK  
036E 500 :+++  
036E 501 ::  
036E 502 :: SHOW\_ONE\_LOCK  
036E 503 ::  
036E 504 :: PURPOSE  
036E 505 :: ACTION ROUTINE TO PROCESS "SHOW LOCK lockid" COMMAND  
036E 506 ::  
036E 507 :: INPUT  
036E 508 :: LOCKID - INDEX INTO LOCK ID TABLE  
036E 509 ::  
036E 510 :: OUTPUT  
036E 511 :: DISPLAYED DATA  
036E 512 ::  
036E 513 ::---  
0000 036E 514 .ENTRY SHOW\_ONE\_LOCK,^M<>  
0370 515  
0000043'EF 00 FB 0370 516 CALLS #0,GET\_LKB : GET LOCK BLOCK ADDRESS  
21 50 E9 0377 517 BLBC R0,10\$ : IF LBC, NO LOCK BLOCK FOR ID  
000004B7'EF 00 FB 037A 518  
FC7B CF 9F 0381 519 CALLS #0,SAVE\_LOCK\_DATA : SAVE LOCK DATA  
00000000'EF 01 FB 0385 520 PUSHAB LOCK HEAD : SET UP LOCK HEADING  
0000051D'EF 00 FB 038C 521 CALLS #1,SET\_HEADING  
0393 522 CALLS #0,DISPLAY\_LOCK : DISPLAY LOCK INFO  
04 039A 523 STATUS SUCCESS : SUCCESSFUL COMPLETION  
039B 524 RET  
04 039B 525  
04 03AD 526 10\$: SIGNAL 0,NOLKB : NO LOCK BLOCK FOR THIS ID  
RET 527

03AE 529 .SBTTL SHOW\_PROC\_LOCK - SHOW LOCKS ASSOCIATED WITH A PROCESS  
 03AE 530 :+++  
 03AE 531 :  
 03AE 532 :  
 03AE 533 :  
 03AE 534 :  
 03AE 535 :  
 03AE 536 :  
 03AE 537 :  
 03AE 538 :  
 03AE 539 :  
 03AE 540 :  
 03AE 541 :  
 03AE 542 :  
 03AE 543 :  
 03AE 544 :---

PURPOSE  
 SHOW LOCKS ASSOCIATED WITH A GIVEN LOCK. CALLED FROM  
 GENERIC CODE WHICH DISPLAYS PROCESS DATA.

INPUT  
 4(AP) - ADDRESS OF PCB COPY IN SDA IMAGE  
 8(AP) - ADDRESS OF "REAL" PCB IN SYSTEM

OUTPUT  
 DISPLAYED DATA

000C 03AE 545 .ENTRY SHOW\_PROC\_LOCK,"M<R2,R3>  
 03B0 546  
 53 04 AC DO 03B0 547 MOVL 4(AP),R3 : GET ADDRESS OF PCB SAVE AREA  
 53 0104 C3 DO 03B4 548 MOVL PCBSL\_LOCKQFL(R3),R3 : GET FORWARD LINK  
 52 08 BC DO 03B9 549 MOVL @8(AP),R2 : GET "REAL" ADDRESS OF PCB  
 52 0104 C2 DE 03BD 550 MOVAL PCBSL\_LOCKQFL(R2),R2 : GET "REAL" ADDRESS OF HEAD OF QUEUE  
 52 53 D1 03C2 551 CMPL R3,R2 : ARE THERE ELEMENTS IN THIS QUEUE?  
 69 13 03C5 552 BEQL NOLOCK : IF EQL, NO

000002BF'EF 53 00000040 8F C3 03C7 553 GETLCK: SUBL3 #LKBSL\_OWNQFL,R3,LKB : ADJUST FOR START OF LKB  
 000004B7'EF 00 FB 03D3 554 CALLS #0,SAVE\_LOCK\_DATA : SAVE LOCK DATA  
 03DA 555 SKIP PAGE  
 03E1 556 PRINT 0,<Lock data:>  
 00000556'EF 00 FB 03EE 557 CALLS #0,FORMAT\_LOCK : FORMAT LOCK DATA  
 03F5 558 ENSURE 9  
 040D 559 SKIP 1  
 00000856'EF 00 FB 0416 560 CALLS #0,PRINT\_LOCK : DISPLAY LOCK INFO  
 52 53 D1 041D 561 REQMEM (R3),R3 : GET NEXT QUEUE LINK  
 14 13 0429 562 CMPL R3,R2 : AT END OF QUEUE?  
 97 11 042C 563 BEQL DONE : IF EQL, DONE  
 042E 564 BRB GETLCK : LOOP  
 0430 565  
 0430 566  
 0430 567 NOLCK:  
 0430 568 SIGNAL 0,NOPRLOCK : THIS PROCESS HAS NO LOCKS  
 0442 569  
 04 0442 570 DONE: RET : SUCCESSFUL

```

0443 572 .SBTTL GET_LKB - GET LOCK BLOCK
0443 573 :*** GET_LKB
0443 574 :
0443 575 :
0443 576 :
0443 577 PURPOSE
0443 578 GIVEN LOCK ID, GET ADDRESS OF LOCK BLOCK.
0443 579 :
0443 580 INPUT
0443 581 LOCKID - INDEX INTO LOCK ID TABLE
0443 582 :
0443 583 OUTPUT
0443 584 R0 - IF LBS, LKB CONTAINS ADDRESS OF LOCK BLOCK
0443 585 IF LBC, NO LOCK BLOCK FOR THIS ID
0443 586 :
0443 587 :--- .ENTRY GET_LKB,"M<R2,R3>
000C 588 :
0445 589 :
0445 590 :
0445 591 VALIDATE INPUT
0445 592 :
52 00000090'EF 3C 0445 593 MOVZWL LOCKID,R2 : MOVE TO REGISTER AND TEST
43 13 044C 594 BEQL 20$ : IF EQL, NOT VALID
      52 D1 044E 595 REQMEM @LOCK$GL_MAXID,R3 : GET MAX ID VALUE
      41 1A 045E 596 CMPL R2,R3 : CHECK IT
      53 6342 DE 0461 597 BGTRU 30$ : IF GTRU, TOO BIG
000002BF'EF 50 D4 0463 598 :
02 18 0483 599 :
050 D6 048E 600 : GET ADDRESS OF LOCK BLOCK
0490 601 :
53 6342 DE 0463 602 REQMEM @LOCK$GL_IDTBL,R3 : GET START OF TABLE
0473 603 MOVAL (R3)[R2],R3 : CALC SLOT ADDRESS
0477 604 REQMEM (R3),R3 : GET LKB ADDRESS
0483 605 CLRL R0 : ASSUME EMPTY SLOT
0485 606 MOVL R3,LKB : MOVE DATA AND TEST
048C 607 BGEQ 10$ : IF GEQ, NO LKB FOR THIS ID
048E 608 INCL R0 : INDICATE SUCCESS
0490 609 :
0490 610 10$: RET : DONE
0491 611 :
0491 612 20$: SIGNAL 0,LOCKIDZER : LOCK ID 0
04 A3 613 RET :
04 A4 614 :
04 A4 615 30$: SIGNAL 0,OUTOFRANG : LOCK ID GREATER THAN MAXID
04 04B6 616 RET :

```

0487 618 .SBTTL SAVE\_LOCK\_DATA - RETRIEVE LOCK DATA FROM SYSTEM  
0487 619 :\*\*\*  
0487 620 :  
0487 621 :  
0487 622 :  
0487 623 :  
0487 624 :  
0487 625 :  
0487 626 :  
0487 627 :  
0487 628 :  
0487 629 :  
0487 630 :  
0487 631 :  
0487 632 :  
0487 633 :  
0487 634 :  
0487 635 :---  
000C 0487 636 .ENTRY SAVE\_LOCK\_DATA,^M<R2,R3>  
0489 637  
52 000002BF'EF DO 04B9 638 MOVL LKB\_R2 : STORE LKB DATA  
53 00000050'EF DO 04C0 639 REQMEM (R2),LKB\_BFR,#LKB\$K\_LENGTH :  
0D 13 04D5 640 MOVL LKB\_BFR+[KB\$L\_PARENT],R3 : PARENT LKB ADDR  
04DC 641 BEQL 10\$ : IF EQ, NO PARENT  
04DE 642 REQMEM LKB\$L\_LKID(R3),R3 : FETCH PARENT LOCK ID  
04EB 643  
00000004'EF 53 DO 04EB 644 10\$: MOVL R3,PARID\_BFR : STORE IN BUFFER  
53 00000058'EF DO 04F2 645 MOVL LKB\_BFR+[KB\$L\_RSB],R3 : GET ASSOCIATED RSB  
04F9 646  
04F9 647 ASSUME RSBSB\_RMOD-RSBSW\_GROUP EQ 2 : MAKE SURE NOTHING CHANGES  
04F9 648 ASSUME RSBSB\_RSNLEN-RSBSW\_GROUP EQ 3  
04F9 649 ASSUME RSBS\$T\_RESNAM-RSBSW\_GROUP EQ 4  
04F9 650 ASSUME RSBSK\_MAXLEN EQ <^X1F>  
04F9 651  
04F9 652 REQMEM RSBSW\_GROUP(R3),LKB\_RSB\_BFR,- : STORE RSB DATA  
04F9 653 #RSBSK\_MAXLEN+4  
050B 654 REQMEM RSBS\$L\_CSID(R3),RSBCSID\_BFR : GET CSID IN RSB  
051C 655  
04 051C 656 RET

051D 658 .SBTTL DISPLAY\_LOCK - CONTROL DISPLAY OF LOCK DATA  
051D 659 :\*\*\*  
051D 660 :  
051D 661 : DISPLAY\_LOCK  
051D 662 :  
051D 663 :  
051D 664 : PURPOSE CONTROLS FORMAT AND DISPLAY OF LOCK DATA  
051D 665 :  
051D 666 : INPUT LOCK DATA AREAS (IMPLIED)  
051D 667 :  
051D 668 :  
051D 669 : OUTPUT DISPLAYED LOCK INFORMATION  
051D 670 :  
051D 671 :  
051D 672 :---  
0000 051D 673 .ENTRY DISPLAY\_LOCK,^M<>  
051F 674  
00000556'EF 00 FB 051F 675 CALLS #0,FORMAT\_LOCK ; FORMAT LOCK DATA  
0526 676 SKIP PAGE  
052D 677 ENSURE 10  
00000856'EF 00 FB 0545 678 SKIP 1  
054E 679 CALLS #0,PRINT\_LOCK ; DISPLAY LOCK DATA  
0555 680  
04 0555 681 RET

0556 683 .SBTTL FORMAT\_LOCK - FORMAT LOCK DATA  
 0556 684 :\*\*\*  
 0556 685 :  
 0556 686 :  
 0556 687 :  
 0556 688 :  
 0556 689 :  
 0556 690 :  
 0556 691 :  
 0556 692 :  
 0556 693 :  
 0556 694 :  
 0556 695 :  
 0556 696 :  
 0556 697 :  
 0556 698 :  
 0556 699 :  
 0556 700 :---  
 0556 701 :  
 007C 0556 702 .ENTRY FORMAT\_LOCK,"M<R2,R3,R4,R5,R6>  
 0558 703 : Lock id., parent id., sublocks and PID  
 0558 704 :  
 0558 705 :  
 56 00000008'EF 9E 0558 706 MOVAB LKB\_BFR,R6  
 0000026F'EF 30 A6 D0 055F 707 MOVL LKB\$L\_LKID(R6),LKID  
 00000283'EF 00000004'EF D0 0567 708 MOVL PARID\_BFR,PARID  
 0000029F'EF 4C A6 3C 0572 709 MOVZWL LKB\$W\_REFCNT(R6),SUBLKS  
 00000273'EF 0C A6 D0 057A 710 MOVL LKB\$L\_PID(R6),PID  
 0582 711 : Lock state and mode(s)  
 36 A6 95 0582 712 :  
 37 19 0585 713 :  
 26 14 0587 714 TSTB LKB\$B\_STATE(R6)  
 00000287'EF FC79 CF 9E 0589 715 BLSS 20\$: : Waiting  
 000002A3'EF FC65 CF 9E 0592 716 BGTR 10\$: : Granted  
 000002AB'EF 50 35 A6 FC21 CF 40 9A 0598 717 MOVAB CV\_STRING,STATE1  
 50 34 A6 9A 05A9 718 MOVAB GR\_STRING,STATE2  
 2E 11 05AD 719 MOVZBL LKB\$B\_GRMODE(R6),R0 : Get granted mode  
 00000287'EF FC48 CF 9E 05AF 720 MOVAW LOCKMODE\_TBL[R0],MODE2 : Store granted mode sting  
 50 35 A6 9A 05B8 721 MOVZBL LKB\$B\_RQMODE(R6),R0 ; Get requested mode  
 BRB 30\$: :  
 00000287'EF 50 35 A6 9A 05BC 722 MOVAB GR\_STRING,STATE1  
 0D 11 0588 723 10\$: MOVZBL LKB\$B\_GRMODE(R6),R0  
 00000287'EF FC52 CF 9E 05BE 724 BRB 25\$: :  
 50 34 A6 9A 05C7 725 20\$: MOVAB WT\_STRING,STATE1  
 000002A3'EF FC53 CF 9E 05CB 726 25\$: MOVZBL LKB\$B\_RQMODE(R6),R0 : No state 2  
 000002AB'EF FC48 CF 9E 05D4 727 MOVAB NULL\_CSTRING,STATE2 : No mode 2  
 0000028F'EF FBE3 CF 40 3E 05DD 728 30\$: MOVAB BLANKS,MODE2 : Store mode 1  
 05E7 729 MOVAW LOCKMODE\_TBL[R0],MODE1  
 05E7 730 30\$: :  
 05E7 731 : Blocking AST  
 05E7 732 :  
 05E7 733 :  
 000002BB'EF FC37 CF 9E 05E7 734 MOVAB NULL\_CSTRING,BLKAST : Assume no blocking AST  
 20 A6 D5 05F0 735 TSTL LKB\$C\_BLKASTADR(R6) : Is there a blocking AST?  
 09 13 05F3 736 BEQL 32\$: :  
 000002BB'EF FC2A CF 9E 05F5 737 MOVAB BL\_STRING,BLKAST : Yes, store address of string  
 05FE 738 : Flags  
 05FE 739 :



|                |                |      |      |        |   |                                 |                               |  |
|----------------|----------------|------|------|--------|---|---------------------------------|-------------------------------|--|
|                | 06 52          | D1   | 06EE | 797    | 40\$:                                     | CMPL R2, #6                     |                               |  |
| 54             | 00000313'EF    | 12   | 06F1 | 798    |   | BNEQ 60\$                       | : Move to third line          |  |
|                | D3 53 09       | DE   | 06F3 | 799    |   | MOVAL STS3, R4                  | : No                          |  |
|                |                | F2   | 06FA | 800    | 60\$:                                     | A0BLSS #9, R3, 358              | : Yes                         |  |
|                |                |      | 06FE | 801    |   |                                 | : Repeat 9 times              |  |
|                |                |      | 06FE | 802    |   |                                 |                               |  |
|                |                |      | 06FE | 803    |   |                                 |                               |  |
|                |                |      | 06FE | 804    |   |                                 |                               |  |
|                |                |      | 06FE | 805    |   |                                 |                               |  |
|                |                |      | 06FE | 806    |   |                                 |                               |  |
| 000002DF'EF    | 0000006F'EF    | 9A   | 06FE | 807    |   | MOVZBL RSNLEN_BFR, RLEN         |                               |  |
| 51             | 0000006F'EF    | 9A   | 0709 | 808    |   | MOVZBL RSNLEN_BFR, R1           |                               |  |
| 50             | 00000070'EF    | 9E   | 0710 | 809    |   | MOVAB RESHAM_BFR, RO            |                               |  |
| 60 20 00 60    | 51             | 50   | DD   | 0717   |   | PUSHL RO                        |                               |  |
|                |                |      | BED0 | 0719   |   | MOVCS R1, (RO), #0, #32, (RO)   | : Save R0                     |  |
|                |                |      |      | 810    |   | POPL RO                         | : Zero out rest of buffer     |  |
|                | 000002CF'EF    | 60   | DE   | 0722   |   | MOVAL (RO), DMP1                | : Restore R0                  |  |
|                | 000002C7'EF    | 80   | DO   | 0729   |   | MOVL (RO)+, RE51                |                               |  |
|                | 000002C3'EF    | 80   | DO   | 0730   |   | MOVL (RO)+, RE52                |                               |  |
|                | 000002EF'EF    | 60   | DE   | 0737   |   | MOVAL (RO), DMP3                |                               |  |
|                | 000002E7'EF    | 80   | DO   | 073E   |   | MOVL (RO)+, RE53                |                               |  |
|                | 000002E3'EF    | 80   | DO   | 0745   |   | MOVL (RO)+, RE54                |                               |  |
|                | 0000030F'EF    | 60   | DE   | 074C   |   | MOVAL (RO), DMP5                |                               |  |
|                | 00000307'EF    | 80   | DO   | 0753   |   | MOVL (RO)+, RE55                |                               |  |
|                | 00000303'EF    | 80   | DO   | 075A   |   | MOVL (RO)+, RE56                |                               |  |
|                | 0000032F'EF    | 60   | DE   | 0761   |   | MOVAL (RO), DMP7                |                               |  |
|                | 00000327'EF    | 80   | DO   | 0768   |   | MOVL (RO)+, RE57                |                               |  |
|                | 00000323'EF    | 80   | DO   | 076F   |   | MOVL (RO)+, RE58                |                               |  |
|                | 50 0000006E'EF | 9A   | 0776 | 824    |   | MOVZBL RMOD_BFR, RO             |                               |  |
| 000002FF'EF    | FB96 CF40      | DO   | 077D | 825    |   | ACMODE TBL[RO], ACMODE          | : Get access mode             |  |
|                | 000001BA'EF    | B5   | 0787 | 826    |   | TSTW GROUP_BUF                  | : System names have group = 0 |  |
|                |                | 44   | 13   | 078D   |   | BEQL 70\$                       | : Branch if system name       |  |
|                |                |      |      | 078F   |   | SFAO_S CTRSTR = FAO GROUP_DSC,- |                               |  |
|                |                |      |      | 078F   |   | OUTBUF = GROUP_BUF_DSC,-        |                               |  |
|                |                |      |      | 078F   |   | P1 = GROUP_BFR                  |                               |  |
|                | 03 50          | E8   | 07AA | 831    |   | BLBS R0, 65\$                   |                               |  |
|                | 00A5           | 31   | 07AD | 832    |   | BRW 90\$                        |                               |  |
| 000001E8'EF    | 000001BA'EF    | 80   | 07B0 | 833    | 65\$:                                     | MOVW GROUP_BUF, GROUP_NUM       |                               |  |
| 000001EA'EF    | 000001BC'EF    | 90   | 07BB | 834    |   | MOVB GROUP_BUF+2, GROUP_NUM+2   |                               |  |
| 0000031F'EF    | 000001D8'EF    | 7E   | 07C6 | 835    |   | MOVAQ GROUP_DSC, NSPACE         |                               |  |
| 0000031F'EF    | 000001C8'EF    | 08   | 11   | 07D1   |   | BRB 80\$                        |                               |  |
|                |                | 7E   | 07D3 | 837    | 70\$:                                     | MOVAQ SYS_DSC, NSPACE           |                               |  |
|                |                |      |      | 07DE   |   |                                 |                               |  |
|                |                |      |      | 07DE   |   |                                 |                               |  |
|                |                |      |      | 07DE   |   |                                 |                               |  |
|                |                |      |      | 07DE   |   |                                 |                               |  |
|                |                |      |      | 07DE   |   |                                 |                               |  |
|                |                |      |      | 080\$: |   |                                 |                               |  |
|                |                |      |      |        |   |                                 |                               |  |
| 50 00000068'EF | 19             | 12   | 07E5 | 841    |   | MOVL RSBCSID_BFR, RO            | : Get CSID                    |  |
| 1F 2A A6 04    | EO             | 07E7 | 842  |        | BNEQ 82\$                                 | : Must be process copy          |                               |  |
| 00000333'EF    | FACA CF        | 9E   | 07EC | 843    | BBS #LKBSV_MSTCPY, LKBSU_STATUS(R6), 84\$ | : Branch if master copy         |                               |  |
| 00000337'EF    | FAD6 CF        | 9E   | 07F5 | 844    | MOVAB LOCAL_TYPE                          | : Set type of lock              |                               |  |
|                | 52             | 11   | 07FE | 845    | MOVAB SPACE, RMINFO                       | : Display no remote info        |                               |  |
|                |                |      |      | 846    | BRB 88\$                                  |                                 |                               |  |
| 00000333'EF    | FABC CF        | 9E   | 0800 | 847    | MOVAB PROCESS, TYPE                       | : Set type of lock              |                               |  |
|                | 0D             | 11   | 0809 | 848    | BRB 86\$                                  |                                 |                               |  |
| 00000333'EF    | FAB9 CF        | 9E   | 0808 | 849    |   |                                 |                               |  |
|                | 50 58 A6       | DO   | 0814 | 850    | MOVAB MASTER_TYPE                         | : Set type of lock              |                               |  |
|                |                |      |      | 851    | MOVL LKBSL_CSID(R6), RO                   | : Get CSID                      |                               |  |
|                |                |      |      | 852    |   |                                 |                               |  |
|                |                |      |      | 853    |   |                                 |                               |  |

51 54 A6 D0 0818 854 868: MOVL LKB\$L, REHLKID(R6), R1 ; Get remote lock fd.  
081C 855 SFAO\_5 CTRSTR = FAO RMINFO\_DSC,-  
081C 856 OUTBUF = RMINFO\_DSC,-  
081C 857 OUTLEN = RMINFO\_LEN,-  
081C 858 P1 = R1,-  
081C 859 P2 = R0  
000000C4'EF 19 50 E9 0839 860 BLBC R0, 90\$  
000000B8'EF 90 083C 861 MOVB RMINFO\_LEN, RMINFO\_CNT  
00000337'EF 9E 0847 862 MOVAB RMINFO\_CNT, RMINFO-  
50 01 D0 0852 864 888: MOVL #1, R0  
04 0855 865 908: RET ; Store ptr to .ASCII string

0856 867 .SBTTL PRINT\_LOCK - OUTPUT LOCK DATA  
 0856 868 :\*\*\*  
 0856 869 :  
 0856 870 :  
 0856 871 :  
 0856 872 :  
 0856 873 :  
 0856 874 :  
 0856 875 :  
 0856 876 :  
 0856 877 :  
 0856 878 :  
 0856 879 :  
 0856 880 :  
 0856 881 :---  
 0000 0856 882 :.ENTRY PRINT\_LOCK,"M<>  
 0858 883 :  
 0858 884 :  
 0858 885 :  
 0858 886 :  
 0000027F'EF DD 0858 887 PUSHL FLAGS1+8  
 0000027B'EF DD 085E 888 PUSHL FLAGS1+4  
 00000277'EF DD 0864 889 PUSHL FLAGS1  
 00000273'EF DD 086A 890 PUSHL PID  
 0000026F'EF DD 0870 891 PUSHL LKID  
 0876 892 PRINT 5,<Lock id: !XL PID: !XL Flags: !3(8AC)>  
 0883 893 :  
 0883 894 :  
 0883 895 :  
 0883 896 :  
 0883 897 :  
 00000298'EF DD 0883 898 PUSHL FLAGS2+8  
 00000297'EF DD 0889 899 PUSHL FLAGS2+4  
 00000293'EF DD 088F 900 PUSHL FLAGS2  
 0000028F'EF DD 0895 901 PUSHL MODE1  
 02 DD 0898 902 PUSHL #2  
 00000287'EF DD 089D 903 PUSHL STATE1  
 00000283'EF DD 08A3 904 PUSHL PARID  
 08A9 905 PRINT 7,<Par. id: !XL !13AC !AD!12\* !3(8AC)>  
 08B6 906 : PARID STATE1 MODE1 FLAGS2  
 08B6 907 :  
 08B6 908 :  
 08B6 909 :  
 08B6 910 :  
 000002B7'EF DD 08B6 911 PUSHL FLAGS3+8  
 000002B3'EF DD 08BC 912 PUSHL FLAGS3+4  
 000002AF'EF DD 08C2 913 PUSHL FLAGS3  
 000002AB'EF DD 08C8 914 PUSHL MODE2  
 02 DD 08CE 915 PUSHL #2  
 000002A3'EF DD 08D0 916 PUSHL STATE2  
 0000029F'EF DD 08D6 917 PUSHL SUBLKS  
 08DC 918 PRINT 7,<Subblocks: !7UL !13AC !AD!12\* !3(8AC)>  
 08E9 919 : SUBLKS STATE2 MODE2 FLAGS3  
 08E9 920 :  
 08E9 921 :  
 08E9 922 :  
 08E9 923 :  
 LINE 3A

LOCK AND RESOURCE FORMATTING ROUTINES  
PRINT\_LOCK - OUTPUT LOCK DATAL 6  
16-SEP-1984 01:31:38 VAX/VMS Macro V04-00  
5-SEP-1984 03:32:46 [SDA.SRC]LOCK.MAR;1Page 24  
(13)

|             |    |      |     |        |              |  |  |  |  |
|-------------|----|------|-----|--------|--------------|--|--|--|--|
| 000002BB'EF | DD | 08E9 | 924 | PUSHL  | BLKAST       |  |  |  |  |
| 000002BF'EF | DD | 08EF | 925 | PUSHL  | LKB          |  |  |  |  |
|             |    | 08F5 | 926 | PRINT  | 2,<LKB:      |  |  |  |  |
|             |    | 0902 | 927 |        |              |  |  |  |  |
|             |    | 0902 | 928 |        |              |  |  |  |  |
|             |    | 0902 | 929 |        |              |  |  |  |  |
|             |    | 0902 | 930 |        |              |  |  |  |  |
|             |    | 0902 | 931 |        |              |  |  |  |  |
|             |    |      |     |        | LINE 4       |  |  |  |  |
| 000002D8'EF | DD | 0902 | 932 | PUSHL  | STS1+8       |  |  |  |  |
| 000002D7'EF | DD | 0908 | 933 | PUSHL  | STS1+4       |  |  |  |  |
| 000002D3'EF | DD | 090E | 934 | PUSHL  | STS1         |  |  |  |  |
| 000002CF'EF | DD | 0914 | 935 | PUSHL  | DMP1         |  |  |  |  |
| 08          | DD | 091A | 936 | PUSHL  | #8           |  |  |  |  |
| 000002C7'EF | DD | 091C | 937 | PUSHL  | RES1         |  |  |  |  |
| 000002C3'EF | DD | 0922 | 938 | PUSHL  | RES2         |  |  |  |  |
|             |    | 0928 | 939 | PRINT  | 7,<Resource: |  |  |  |  |
|             |    | 0935 | 940 |        |              |  |  |  |  |
|             |    | 0935 | 941 |        |              |  |  |  |  |
|             |    | 0935 | 942 |        |              |  |  |  |  |
|             |    | 0935 | 943 |        |              |  |  |  |  |
|             |    | 0935 | 944 |        |              |  |  |  |  |
|             |    |      |     | LINE 5 |              |  |  |  |  |
| 000002FB'EF | DD | 0935 | 945 | PUSHL  | STS2+8       |  |  |  |  |
| 000002F7'EF | DD | 0938 | 946 | PUSHL  | STS2+4       |  |  |  |  |
| 000002F3'EF | DD | 0941 | 947 | PUSHL  | STS2         |  |  |  |  |
| 000002EF'EF | DD | 0947 | 948 | PUSHL  | DMP3         |  |  |  |  |
| 08          | DD | 094D | 949 | PUSHL  | #8           |  |  |  |  |
| 000002E7'EF | DD | 094F | 950 | PUSHL  | RES3         |  |  |  |  |
| 000002E3'EF | DD | 0955 | 951 | PUSHL  | RES4         |  |  |  |  |
| 000002DF'EF | DD | 0958 | 952 | PUSHL  | RLEN         |  |  |  |  |
|             |    | 0961 | 953 | PRINT  | 8,< Length   |  |  |  |  |
|             |    | 096E | 954 |        | !2B          |  |  |  |  |
|             |    | 096E | 955 |        | RLEN         |  |  |  |  |
|             |    | 096E | 956 |        | RES4         |  |  |  |  |
|             |    | 096E | 957 |        | RES3         |  |  |  |  |
|             |    | 096E | 958 |        | DMP3         |  |  |  |  |
|             |    |      |     | LINE 6 |              |  |  |  |  |
| 0000031B'EF | DD | 096E | 959 | PUSHL  | STS3+8       |  |  |  |  |
| 00000317'EF | DD | 0974 | 960 | PUSHL  | STS3+4       |  |  |  |  |
| 00000313'EF | DD | 097A | 961 | PUSHL  | STS3         |  |  |  |  |
| 0000030F'EF | DD | 0980 | 962 | PUSHL  | DMP5         |  |  |  |  |
| 08          | DD | 0986 | 963 | PUSHL  | #8           |  |  |  |  |
| 00000307'EF | DD | 0988 | 964 | PUSHL  | RES5         |  |  |  |  |
| 00000303'EF | DD | 098E | 965 | PUSHL  | RES6         |  |  |  |  |
| 000002FF'EF | DD | 0994 | 966 | PUSHL  | ACMODE       |  |  |  |  |
|             |    | 099A | 967 | PRINT  | 8,< !11AC    |  |  |  |  |
|             |    | 09A7 | 968 |        | ACMODE       |  |  |  |  |
|             |    | 09A7 | 969 |        | RES6         |  |  |  |  |
|             |    | 09A7 | 970 |        | RES5         |  |  |  |  |
|             |    | 09A7 | 971 |        | DMP5         |  |  |  |  |
|             |    | 09A7 | 972 |        |              |  |  |  |  |
|             |    |      |     | LINE 7 |              |  |  |  |  |
| 0000032F'EF | DD | 09A7 | 973 | PUSHL  | DMP7         |  |  |  |  |
| 08          | DD | 09AD | 974 | PUSHL  | #8           |  |  |  |  |
| 00000327'EF | DD | 09AF | 975 | PUSHL  | RES7         |  |  |  |  |
| 00000323'EF | DD | 09B5 | 976 | PUSHL  | RES8         |  |  |  |  |
| 0000031F'EF | DD | 09BB | 977 | PUSHL  | NSPACE       |  |  |  |  |
|             |    | 09C1 | 978 | PRINT  | 5,< !11AS    |  |  |  |  |
|             |    | 09CE | 979 |        | NSPACE       |  |  |  |  |
|             |    | 09CE | 980 |        | RES8         |  |  |  |  |
|             |    |      |     |        | RES7         |  |  |  |  |
|             |    |      |     |        | DMP7         |  |  |  |  |

LOCK  
V04-000

LOCK AND RESOURCE FORMATTING ROUTINES  
PRINT\_LOCK - OUTPUT LOCK DATA

M 6

16-SEP-1984 01:31:38 VAX/VMS Macro V04-00  
5-SEP-1984 03:32:46 [SDA.SRC]LOCK.MAR;1

Page 25  
(13)

00000337'EF 09CE 981 :  
00000333'EF DD 09CE 982 : LINE 8  
00000333'EF DD 09D4 983 :  
09DA 984 PUSHL RMINFO  
09E7 985 PUSHL TYPE  
04 09E7 986 PRINT 2.<!AC copy !AC!/>  
09E7 987 : TYPE RMINFO  
04 09E7 988 RET

09E8 990 .SBTTL LOCK\_COND\_HAND - CONDITION HANDLER FOR SHOW\_ALL\_LOCKS  
 09E8 991 :+++  
 09E8 992 :  
 09E8 993 :  
 09E8 994 :  
 09E8 995 :  
 09E8 996 : PROVIDE EXIT PATH FOR SHOW\_ALL\_LOCKS WHEN THERE ARE NO MORE LOCKS  
 09E8 997 : TO BE PROCESSED  
 09E8 998 :  
 09E8 999 :  
 09E8 1000 :  
 09E8 1001 :  
 09E8 1002 :  
 09E8 1003 :  
 09E8 1004 :  
 09E8 1005 :  
 09E8 1006 :  
 09E8 1007 :---  
 001C 09E8 1008 .ENTRY LOCK\_COND\_HAND,^M<R2,R3,R4>  
 09EA 1009

|                |              |                                   |  |
|----------------|--------------|-----------------------------------|--|
| 52 04 AC 7D    | 09EA 1010    | MOVQ 4(AP),R2                     | : GET ADDRESSES OF ARRAYS                  |
| 54 04 A2 D0    | 09EE 1011    | MOVL CHFSL SIG NAME(R2),R4        | : GET CONDITION NAME                       |
| 54 00000000'8F | D1 09F2 1012 | CMPL #SSS_ONWIND,R4               | : ARE WE UNWINDING?                        |
| 31 13          | 09F9 1013    | BEQL 20\$                         | : IF EQL, YES                              |
| 54 00000000'8F | D1 09FB 1014 | CMPL #MSG\$_OUTOFRANG,R4          | : DID WE GET THE SIGNAL WE'RE LOOKING FOR? |
| 28 12          | 0A02 1015    | BNEQ 20\$                         | : IF NEQ, NO - PASS IT ON                  |
| 00000000'EF    | 0A04 1016    | TSTL LOCK_COUNT                   | : DID WE PROCESS LOCKS?                    |
| 12 12          | 0A04 1017    | BNEQ 10\$                         | : IF NEQ, YES                              |
|                | 0A0A 1018    | SIGNAL 0,NOLOCKS                  | : DISPLAY INFO MESSAGE                     |
| OC A3 50       | 0A1E 1020    |                                   |  |
| 00000000'GF    | D0 0A1E 1021 | 10\$: MOVL R0,CHFSL_MCH_SAVR0(R3) | : INDICATE SUCCESS AFTER UNWIND            |
| 02 7E          | 0A22 1022    | CLRQ -(SP)                        | : GO BACK TO ESTABLISHER                   |
|                | FB 0A24 1023 | CALLS #2,G^SYSSUNWIND             | : UNWIND CALL FRAMES                       |
|                | 04 0A2B 1024 | RET                               | : RETURN TO ESTABLISHER                    |
| 50 0000'8F     | 3C 0A2C 1025 |                                   |  |
|                | 04 0A31 1026 | 20\$: MOVZWL #SSS_RESIGNAL,RO     | : RESIGNAL CONDITION                       |
|                |              | RET                               |  |

```

0A32 1029 .SBTTL SHOW_ALL_RES - ACTION ROUTINE TO DISPLAY ALL RESOURCES
0A32 1030 ++++
0A32 1031
0A32 1032 SHOW_ALL_RES
0A32 1033
0A32 1034 PURPOSE
0A32 1035 MAIN ROUTINE TO PROCESS THE "SHOW RES/ALL" COMMAND
0A32 1036
0A32 1037 INPUT
0A32 1038 NONE
0A32 1039
0A32 1040 OUTPUT
0A32 1041 DISPLAYED DATA
0A32 1042
0A32 1043 ---+
0A32 1044 .ENTRY SHOW_ALL_RES,^M<R2,R3,R4>
0A32 1045
0A34 1046 PUSHAB RES HEAD ; SET UP HEADING
0A38 1047 CALLS #1,SET HEADING
0A3F 1048 CLRL RES_COUNT ; INIT COUNT
0A45 1049 CLRL R4 ; INIT HASH TABLE INDEX
0A47 1050 REQMEM BLCK$GL HTBL_CNT,R2 ; GET COUNT OF ELEMENTS IN HASH TABLE
0A57 1051 ASHL R2,#1,HTBL_CNT
0A5F 1052 MOVL LCK$GL HASHTBL,R3 ; GET ADDRESS OF START OF HASH TABLE
0A66 1053 REQMEM (R3),R3 ; GET FIRST ELEMENT
0A72 1054
0A72 1055 : INDEX THROUGH HASH TABLE, LOOKING FOR HASH CHAINS
0A72 1056
0A72 1057 10$: :
0A72 1058 MOVAL (R3)[R4],R2 ; GET HEAD OF CHAIN
0A72 1059 REQMEM (R2),RSB ; GET FIRST RSB ADDRESS
0A86 1060 TSTL RSB ; IF ZERO, EMPTY CHAIN
0A8C 1061 BEQL 20$ ; IF EQL, NO
0A8E 1062
0A8E 1063
0A8E 1064 : SEQUENCE THROUGH HASH CHAIN
0A8E 1065
0A8E 1066
0A8E 1067 15$: :
0A8E 1068 INCL RES_COUNT ; INCREMENT COUNT
0A94 1069 REQMEM @RSB,RSB BFR,#RSB$K_MAXLEN+RSBSK_LENGTH ;SAVE RSB DATA
0AAD 1070 CALLS #0,SHOW_RSB ; DISPLAY RSB
0AB4 1071 CALLS #0,SHOW_QUEUES ; DISPLAY QUEUES
0ABB 1072 MOVL RSB-BFR,RSB ; GET NEXT RSB
0AC6 1073 BLSS 15$ ; IF EQL, NO
0AC8 1074
0AC8 1075 : OUTER LOOP CONTROL
0AC8 1076
0AC8 1077
0AC8 1078 20$: :
0AC8 1079 A0BLSS HTBL_CNT,R4,10$ ; INCREMENT INDEX, LOOP IF LSS
0ADO 1080
0ADO 1081 TSTL RES_COUNT ; DID WE PROCESS ANY RESOURCES?
0AD6 1082 BEQL 30$ ; IF EQL, NO
0AD8 1083
0AD8 1084 STATUS SUCCESS ; SUCCESS RETURN
0ADF 1085 RET

```

LOCK  
V04-000

C 7  
LOCK AND RESOURCE FORMATTING ROUTINES 16-SEP-1984 01:31:38 VAX/VMS Macro V04-00  
SHOW\_ALL\_RES - ACTION ROUTINE TO DISPLAY 5-SEP-1984 03:32:46 [SDA.SRC]LOCK.MAR;1

Page 28  
(15)

04 OAE0 1086 30\$:  
OAE0 1087 SIGNAL 0,NORESOURC  
OAF2 1088 RET ; NO RESOURCES DISPLAYED

0AF3 1090  
 0AF3 1091 .SBTTL SHOW\_ONE\_RES - ACTION ROUTINE TO DISPLAY ONE RESOURCE  
 0AF3 1092 :+++:  
 0AF3 1093 :  
 0AF3 1094 :  
 0AF3 1095 :  
 0AF3 1096 :  
 0AF3 1097 :  
 0AF3 1098 :  
 0AF3 1099 :  
 0AF3 1100 :  
 0AF3 1101 :  
 0AF3 1102 :  
 0AF3 1103 :  
 0AF3 1104 :  
 0AF3 1105 :---:  
 0004 0AF3 1106 .ENTRY SHOW\_ONE\_RES.^M<R2>  
 0AF5 1107  
 F949 CF 00 FB 0AF5 1108 CALLS #0,GET\_LKB : GET LKB ADDRESS  
 52 50 E9 0AFA 1109 BLBC R0,10\$ : IF LBC, NONE EXISTS  
 000002BF'EF D0 0AFD 1110 MOVL LKB,R2  
 0B04 1111 REQMEM LKB\$L,RSB(R2),RSB : GET ADDRESS OF RSB  
 0B15 1112 REQMEM @RSB,RSB\_BFR,RSB\$K\_MAXLEN+RSB\$K\_LENGTH ; GET RSB DATA  
 F4E3 CF 9F 0B2E 1113 PUSHAB RES\_HEAD : SET RSB HEADING  
 00000000'EF 01 FB 0B32 1114 CALLS #1,SET\_HEADING  
 00000B62'EF 00 FB 0B39 1115 CALLS #0,SHOW\_RSB : DISPLAY RSB  
 00000B92'EF 00 FB 0B40 1116 CALLS #0,S\_JW\_QUEUES : DISPLAY QUEUES  
 0B47 1117  
 0B47 1118 STATUS SUCCESS  
 04 0B4E 1119 RET  
 0B4F 1120  
 04 0B4F 1121 10\$: SIGNAL 0,NOLKB : LOCKID NON-EXISTANT  
 04 0B61 1122 RET

0B62 1124 .SBTTL SHOW\_RSB - CONTROL FOR RSB DISPLAY  
0B62 1125 :+++:  
0B62 1126 ::  
0B62 1127 :  
0B62 1128 :  
0B62 1129 :  
0B62 1130 :  
0B62 1131 :  
0B62 1132 :  
0B62 1133 :  
0B62 1134 :  
0B62 1135 :  
0B62 1136 :  
0B62 1137 :---:  
0000 0B62 1138 .ENTRY SHOW\_RSB,"M<>  
0B64 1139  
00000BCC'EF 00 FB 0B64 1140 CALLS #0,FORMAT\_RSB  
0B68 1141 SKIP PAGE  
00000D34'EF 00 FB 0B72 1142 ENSURE 9  
0B8A 1143 CALLS #0,PRINT\_RSB  
0B91 1144  
04 0B91 1145 RET

0B92 1147 .SBTTL SHOW\_QUEUES - DISPLAY QUEUES FOR GIVEN RESOURCE  
 0B92 1148 ;+++  
 0B92 1149 :  
 0B92 1150 :  
 0B92 1151 :  
 0B92 1152 :  
 0B92 1153 :  
 0B92 1154 :  
 0B92 1155 :  
 0B92 1156 :  
 0B92 1157 :  
 0B92 1158 :  
 0B92 1159 :  
 0B92 1160 :  
 0B92 1161 :  
 0B92 1162 :---  
 0000 0B92 1163 .ENTRY SHOW\_QUEUES,"MC>  
 0B94 1164  
 0B94 1165  
 7E 10 000001EB'EF C1 0B94 1166 ADDL3 RSB,#RSBSL\_GROFL,-(SP) ; GET ADDRESS OF GRANT QUE FLINK  
 FFFFFFFF 8F DD 0B9C 1167 PUSHL #GRANT ; IND GRANT QUE  
 00000E55'EF 02 FB 0BA2 1168 CALLS #2,PROCESS\_QUEUE  
 7E 18 000001EB'EF C1 0BA9 1170 ADDL3 RSB,#RSBSL\_CVTQFL,-(SP) ; GET ADDRESS OF CONVERT QUE FLINK  
 00 00 DD 0B81 1171 PUSHL #CONVERT ; IND CONVERT QUE  
 00000E55'EF 02 FB 0B83 1172 CALLS #2,PROCESS\_QUEUE  
 7E 20 000001EB'EF C1 0BBA 1174 ADDL3 RSB,#RSBSL\_WTQFL,-(SP) ; GET ADDRESS OF WAIT QUE FLINK  
 01 01 DD 0BC2 1175 PUSHL #WAIT ; IND WAIT QUE  
 00000E55'EF 02 FB 0BC4 1176 CALLS #2,PROCESS\_QUEUE  
 04 0BCB 1177  
 RET

MAI  
VOA

; Branch if system name

|             |             |       |      |      |        |                          |                           |
|-------------|-------------|-------|------|------|--------|--------------------------|---------------------------|
| 000001EF'EF | F51B CF41   | D0    | OCB1 | 1237 | MOVL   | LKMODE TBL[R1],GGMOD     |                           |
|             | 51 OD A6    | 9A    | OCBB | 1238 | MOVZBL | RSBSB CGMODE(R6),R1      |                           |
| 000001F7'EF | F50D CF41   | D0    | OCBF | 1239 | MOVL   | LKMODE TBL[R1],CGMOD     |                           |
|             | 51 4E A6    | 90    | OCC9 | 1240 | MOVB   | RSBSB RMOD(R6),R1        |                           |
| 00000243'EF | F646 CF41   | D0    | OCCD | 1241 | MOVL   | ACMODE TBL[R1],RACMOD    |                           |
|             | 0000022B'EF | 4F A6 | 9A   | OCDF | 1242   | MOVZBL                   | RSBSB RSNLEN(R6),RESNLEN  |
| 0000025B'EF | 000001C8'EF | 7E    | OCEA | 1243 | MOVAQ  | SYS DSC,RNSPACE          |                           |
|             | 0000006C'EF | 4C A6 | B0   | OCF2 | 1244   | MOVW                     | RSBSW_GROUP(R6),GROUP_BFR |
|             |             | 3F    | 15   | OCF4 | 1245   | BEQL                     | 90S                       |
|             |             |       |      | OCF4 | 1246   | SFAO_S                   | CTRSTR = FAO_GROUP_DSC,-  |
|             |             |       |      | OCF4 | 1247   |                          | OUTBUF = GROUP_BUF_DSC,-  |
|             |             |       |      | OCF4 | 1248   |                          | P1 = GROUP_BFR            |
| 000001E8'EF | 21 50       | E9    | OD0F | 1249 | BLBC   | R0,90S                   |                           |
| 000001EA'EF | 000001BA'EF | B0    | OD12 | 1250 | MOVW   | GROUP_BUF, GROUP_NUM     |                           |
| 0000025B'EF | 000001BC'EF | 90    | OD1D | 1251 | MOVB   | GROUP_BUF+2, GROUP_NUM+2 |                           |
|             | 000001D8'EF | 7E    | OD28 | 1252 | MOVAQ  | GROUP_DSC,RNSPACE        |                           |
|             |             |       | OD33 | 1253 |        |                          |                           |
|             |             |       | OD33 | 1254 | 90S:   |                          |                           |
|             |             | 04    | OD33 | 1255 | RET    |                          |                           |

0D34 1257 .SBTTL PRINT\_RSB - OUTPUT RSB DATA  
 0D34 1258 :\*\*\*  
 0D34 1259 :  
 0D34 1260 :  
 0D34 1261 :  
 0D34 1262 :  
 0D34 1263 :  
 0D34 1264 :  
 0D34 1265 :  
 0D34 1266 :  
 0D34 1267 :  
 0D34 1268 :  
 0D34 1269 :  
 0D34 1270 :  
 0D34 1271 :---  
 0000 0D34 1272 .ENTRY PRINT\_RSB,"M">  
 0D36 1273 :  
 0D36 1274 :  
 0D36 1275 :  
 0D36 1276 : LINE 1  
 0D36 1277 :  
 000001EF'EF DD 0D36 1278 PUSHL GGMOD  
 000001EB'EF DD 0D3C 1279 PUSHL RSB  
 0D42 1280 PRINT 2,<Address of RSB: !XL Group grant mode:  
 0D4F 1281 : RSB !AC >  
 0D4F 1282 :  
 0D4F 1283 :  
 0D4F 1284 :  
 0D4F 1285 : LINE 2  
 000001F7'EF DD 0D4F 1286 PUSHL CGMOD  
 000001F3'EF DD 0D55 1287 PUSHL PRSB  
 0D5B 1288 PRINT 2,<Parent RSB: !XL Conversion grant mode: !AC  
 0D68 1289 : PRSB >  
 0D68 1290 :  
 0D68 1291 :  
 0D68 1292 :  
 0D68 1293 :  
 000001FF'EF DD 0D68 1294 PUSHL BAST  
 000001FB'EF DD 0D6E 1295 PUSHL SRSBCT  
 0D74 1296 PRINT 2,<Sub-RSB count: !BUL BLKAST count: !BUL>  
 0D81 1297 : SRSBCT BAST  
 0D81 1298 :  
 0D81 1299 :  
 0D81 1300 :  
 0D81 1301 :  
 00000217'EF DD 0D81 1302 PUSHL VALID  
 00000213'EF DD 0D87 1303 PUSHL SEQNUM  
 0000020F'EF DD 0D8D 1304 PUSHL VAL1  
 00000208'EF DD 0D93 1305 PUSHL VAL2  
 00000207'EF DD 0D99 1306 PUSHL VAL3  
 00000203'EF DD 0D9F 1307 PUSHL VAL4  
 0DAS 1308 PRINT 6,<Value block: !XL !XL !XL !XL Seq. #: !XL !AC>  
 0DB2 1309 : VAL4 VAL3 VAL2 VAL1 SEQNUM VALID  
 0DB2 1310 :  
 0DB2 1311 :  
 0DB2 1312 :  
 0DB2 1313 :  
 LINE 5

|                   |      |      |       |              |                          |
|-------------------|------|------|-------|--------------|--------------------------|
| 00000227'EF<br>08 | DD   | 0DB2 | 1314  | PUSHL        | BUF1                     |
| 0000021F'EF       | DD   | 0DB8 | 1315  | PUSHL        | #8                       |
| 0000021B'EF       | DD   | 0DBA | 1316  | PUSHL        | RESN1                    |
|                   | DD   | 0DC0 | 1317  | PUSHL        | RESN2                    |
|                   | ODC6 | 1318 | PRINT | 4,<Resource: | !XL !XL !AF >            |
|                   | ODD3 | 1319 |       |              | RESN2 RESN1 BUF1         |
|                   | ODD3 | 1320 | :     |              |                          |
|                   | ODD3 | 1321 | :     |              |                          |
|                   | ODD3 | 1322 | :     |              |                          |
|                   | ODD3 | 1323 | :     |              |                          |
|                   | ODD3 | 1324 |       |              | LINE 6                   |
| 0000023F'EF       | DD   | ODD3 | 1324  | PUSHL        | CSID                     |
| 0000023B'EF       | DD   | ODD9 | 1325  | PUSHL        | BUF2                     |
| 08                | DD   | ODDF | 1326  | PUSHL        | #8                       |
| 00000233'EF       | DD   | ODE1 | 1327  | PUSHL        | RESN3                    |
| 0000022F'EF       | DD   | ODE7 | 1328  | PUSHL        | RESN4                    |
| 0000022B'EF       | DD   | ODED | 1329  | PUSHL        | RESNLEN                  |
|                   | ODF3 | 1330 | PRINT | 6,< Length   | !2UL !XL !XL !AF         |
|                   | OE00 | 1331 |       |              | RESNLEN RESN4 RESN3 BUF2 |
|                   | OE00 | 1332 | :     |              |                          |
|                   | OE00 | 1333 | :     |              |                          |
|                   | OE00 | 1334 | :     |              |                          |
|                   | OE00 | 1335 | :     |              |                          |
| 00000257'EF       | DD   | OE00 | 1336  | PUSHL        | DIRENTRY                 |
| 00000253'EF       | DD   | OE06 | 1337  | PUSHL        | BUF3                     |
| 08                | DD   | OE0C | 1338  | PUSHL        | #8                       |
| 0000024B'EF       | DD   | OE0E | 1339  | PUSHL        | RESN5                    |
| 00000247'EF       | DD   | OE14 | 1340  | PUSHL        | RESN6                    |
| 00000243'EF       | DD   | OE1A | 1341  | PUSHL        | RACMOD                   |
|                   | OE20 | 1342 | PRINT | 6,< !1AC     | !XL !XL !AF              |
|                   | OE20 | 1343 |       |              | RACMOD RESN6 RESN5 BUF3  |
|                   | OE20 | 1344 | :     |              |                          |
|                   | OE20 | 1345 | :     |              |                          |
|                   | OE20 | 1346 | :     |              |                          |
|                   | OE20 | 1347 | :     |              |                          |
| 00000268'EF       | DD   | OE2D | 1348  | PUSHL        | BUF4                     |
| 08                | DD   | OE33 | 1349  | PUSHL        | #8                       |
| 00000263'EF       | DD   | OE35 | 1350  | PUSHL        | RESN7                    |
| 0000025F'EF       | DD   | OE3B | 1351  | PUSHL        | RESN8                    |
| 0000025B'EF       | DD   | OE41 | 1352  | PUSHL        | RNSPACE                  |
|                   | OE47 | 1353 | PRINT | 5,< !1AS     | !XL !XL !AF >            |
|                   | OE54 | 1354 |       |              | RNSPACE RESN8 RESN7 BUF4 |
|                   | OE54 | 1355 | :     |              |                          |
| 04                | OE54 | 1356 | RET   |              |                          |

.SBTTL PROCESS\_QUEUE - TRAVERSE RESOURCE QUEUES

PROCESS\_QUEUE

PURPOSE  
TRAVERSE AN INDIVIDUAL RESOURCE QUEUE, FORMATTING AND DISPLAYING ITS CONTENTS.

INPUT  
4(AP) - QUEUE TYPE, WHERE: -1: GRANT  
0: CONVERSION  
1: WAIT  
8(AP) - ADDRESS OF HEAD OF QUEUE

OUTPUT  
DISPLAYED CONTENTS OF QUEUE

03FC 0E55 1377 .ENTRY PROCESS\_QUEUE.^M<R2,R3,R4,R5,R6,R7,R8,R9>

0E57 1378

54 000001A2'EF 9E 0E57 1379 MOVAB LKID\_BFR,R4 : GET ADDRESS OF LOCKID STORAGE BUFFER  
 55 0000019C'EF 9E 0E5E 1380 MOVAB GRMD\_BFR,R5 : GET ADDRESS OF GRANT MODE BUFFER  
 56 0000019F'EF 9E 0E65 1381 MOVAB RQMD\_BFR,R6 : GET ADDRESS OF REQUEST MODE BUFFER  
 52 08 AC D0 0E6C 1382 MOVL 8(AP),R2 : GET ADDRESS OF HEAD OF QUEUE  
 59 52 D0 0E70 1383 MOVL R2,R9  
 00000119'EF D4 0E73 1384 CLRL QUEUE\_COUNT : INIT COUNT OF QUEUE ITEMS PROCESSED  
 00000198'EF 04 AC D0 0E79 1385 MOVL 4(AP),QUEUE\_TYPE : GET TYPE OF QUEUE  
 58 01 CE 0E81 1386 MNEG L #1,R8  
 57 00000198'EF 01 C1 0E84 1387 ADDL3 #1,QUEUE\_TYPE,R7  
 57 F1BD CF47 D0 0E8C 1388 MOVL QUE\_STR\_TBL[R7],R7 : GET HEADER FOR THIS TYPE OF QUEUE  
 0E92 1389 ENSURE 3  
 0EAA 1390 SKIP 1  
 0EB3 1391 PRINTD R8,(R7) : DISPLAY IT  
 0EBE 1392  
 0EBE 1393 : INIT FOR SCAN OF QUEUE  
 53 D4 0EBE 1395 10\$: CLRL R3  
 0EC0 1397  
 0EC0 1398 : GET QUEUE ELEMENT  
 0EC0 1400 :  
 52 59 D1 0EC0 1401 20\$: REQMEM (R2),R2 : GET QUEUE ELEMENT  
 30 13 0ECC 1402 CMPL R9,R2 : AT END OF QUEUE?  
 0ECF 1403 BEQL 30\$ : IF EQL, YES  
 0ED1 1404  
 0ED1 1405 ASSUME LKBSDL\_SQFL-LKBSDL\_LKID GT 0  
 0ED1 1406 ASSUME LKBSSB\_RQMODE-LKBSDL\_LKID EQ 4  
 0ED1 1407 ASSUME LKBSSB\_GRMODE-LKBSDL\_LKID EQ 5  
 0ED1 1408  
 00000119'EF D6 0ED1 1409 INCL QUEUE\_COUNT : INCREMENT COUNT  
 58 52 08 C3 0ED7 1410 SUBL3 #<LKBSDL\_SQFL-LKBSDL\_LKID>,R2,R8 : GET START OF RELEVANT DATA  
 00000008'EF 9E 0EDB 1411 REQMEM (R8),LKB\_BFR,#6 : SAVE ID, GRANT & REQUEST MODE  
 6443 67 D0 0EF3 1412 MOVAB LKB\_BFR,R7  
 6543 05 A7 90 0EF7 1413 MOVL LKID\_OFF(R7),(R4)[R3] : MOVE INTO STORAGE BUFFERS  
 0EF7 1414 MOVB GRMD\_OFF(R7),(R5)[R3]

6643 04 A7 90 0EFC 1415  
      0F01 1416  
      F2 0F01 1417  
      FB 0F05 1418  
      B0 11 0FOC 1419  
      0F0E 1420  
      FB 0F0E 1421 30\$:  
00000F2B'EF 00 DS 0F15 1422  
00000119'EF 0D 12 0F1B 1423  
            OF1D 1424  
            OF2A 1425  
04 0F2A 1426 EXIT: RET

MOV B RQMD\_OFF(R7), (R6)[R3]  
AOBLSS #3,R3 20\$ ; IF WE'VE SAVED 3 ITEMS, PRINT THEM  
CALLS #0 PRINT\_LINE  
BRB 10\$ ; GET MORE ITEMS  
CALLS #0 PRINT\_LINE ; PRINT ANY REMAINING ITEMS  
TSTL QUEUE\_COUNT ; WERE ANY ITEMS PROCESSED?  
BNEQ EXIT ; IF NEQ, YES  
PRINT 0,< \*\*\* EMPTY QUEUE \*\*\*>

OF2B 1428 .SBTTL PRINT\_LINE - OUTPUT QUEUE DATA  
 OF2B 1429 :\*\*\*  
 OF2B 1430 :  
 OF2B 1431 :  
 OF2B 1432 :  
 OF2B 1433 :  
 OF2B 1434 :  
 OF2B 1435 :  
 OF2B 1436 :  
 OF2B 1437 :  
 OF2B 1438 :  
 OF2B 1439 :  
 OF2B 1440 :  
 OF2B 1441 :  
 OF2B 1442 :  
 OF2B 1443 :  
 OF2B 1444 :  
 OF2B 1445 :  
 OF2B 1446 :  
 OF2B 1447 :---  
 OF2B 1448 :  
 0180 OF2B 1449 .ENTRY PRINT\_LINE,^M<R7,R8>  
 OF2D 1450 :  
 OF2D 1451 :  
 OF2D 1452 :  
 OF2D 1453 :  
 57 00000198'EF D0 OF2D 1454 MOVL QUEUE\_TYPE,R7 : SAVE QUEUE TYPE  
 00000194'EF 53 D0 OF34 1455 MOVL R3,COUNT : SAVE COUNT  
 01 53 F4 OF3B 1456 S0BGEQ R3,10\$ : CONVERT TO ARRAY INDEX  
 04 OF3E 1457 RET : IF COUNT 0, NO ITEMS TO DISPLAY  
 OF3F 1458 :  
 OF3F 1459 :  
 OF3F 1460 :  
 OF3F 1461 :  
 57 D5 OF3F 1462 10\$: TSTL R7 : ONLY FORMAT RQ MODE IF WAIT OR CONVERT  
 09 19 OF41 1463 BLSS 15\$ : QUEUE  
 58 6643 9A OF43 1464 MOVZBL (R6)[R3],R8 :  
 F285 CF48 DD OF47 1465 PUSHL LKMODE\_TBL[R8]  
 OF4C 1466 :  
 57 D5 OF4C 1467 15\$: TSTL R7 : ONLY FORMAT GR MODE IF GRANT OR CONVERT  
 09 14 OF4E 1468 BGTR 20\$ : QUEUE  
 58 6543 9A OF50 1469 MOVZBL (R5)[R3],R8 :  
 F278 CF48 DD OF54 1470 PUSHL LKMODE\_TBL[R8]  
 OF59 1471 :  
 6443 DD OF59 1472 20\$: PUSHL (R4)[R3] : PROCESS LOCK ID  
 E0 53 F4 OF5C 1473 S0BGEQ R3,10\$ : IF NOT YET DONE, LOOP  
 OF5F 1474 :  
 58 F0DB CF 9E OF5F 1475 MOVAB CONV\_STR\_TBL,R8 : GET ADDRESS OF CORRECT FAO STRING TABLE  
 57 D5 OF64 1476 TSTL R7 :  
 05 13 OF66 1477 BEQL 25\$ : IF EQ, CONVERT  
 58 F0C2 CF 9E OF68 1478 MOVAB LOCK\_STR\_TBL,R8 : ELSE WAIT/GANT  
 OF6D 1479 :  
 53 00000194'EF D0 OF6D 1480 25\$: MOVL COUNT,R3 : PICK UP CORRECT STRING FOR NUMBER OF  
 58 6843 D0 OF74 1481 MOVL (R8)[R3],R8 : ITEMS  
 53 53 C0 OF78 1482 ADDL R3,R3 : CALCULATE CORRECT FAO ARG COUNT  
 57 D5 OF7B 1483 TSTL R7 :  
 07 12 OF7D 1484 BNEQ 30\$ :

53 00000194'EF C0 0F7F 1485 ADDL COUNT,R3  
0F86 1486  
0F86 1487 30\$: PRINTD r3,(r8) : DISPLAY LINE  
04 0F91 1488 RET  
0F91 1489  
0F92 1490  
0F92 1491 .END

|                 |               |    |                  |              |    |
|-----------------|---------------|----|------------------|--------------|----|
| SST2            | = 00000004    |    | GROUP_TXT        | 000001E0 R   | 02 |
| ACMODE          | = 000002FF R  | 02 | GR_STRING        | 000001FB R   | 03 |
| ACMODE_TBL      | = 00000318 R  | 03 | HTBL_CNT         | 00000190 R   | 02 |
| ARGS            | = 00000003    |    | HTBL_INDX        | 0000018C R   | 03 |
| BAST            | = 000001FF R  | 02 | KMODE            | 000002EB R   | 04 |
| BLANKS          | = 00000220 R  | 03 | LCK\$GL_HASHTBL  | ***** X      | 03 |
| BLKAST          | = 000002BB R  | 02 | LCK\$GL-HTBLCNT  | ***** XX     | 03 |
| BL_STRING       | = 00000223 R  | 03 | LCK\$GL-IDTBL    | ***** XX     | 03 |
| BUF1            | = 00000227 R  | 02 | LCK\$GL-MAXID    | ***** XX     | 03 |
| BUF2            | = 00000238 R  | 02 | LIB\$SIGNAL      | ***** XX     | 03 |
| BUF3            | = 00000253 R  | 02 | LINE_COUNT       | ***** XX     | 03 |
| BUF4            | = 00000268 R  | 02 | LKB              | 000002BF R   | 02 |
| CGMOD           | = 000001F7 R  | 02 | LKB\$B_GRMODE    | = 00000035   |    |
| CHFSL_MCH_SAVRO | = 0000000C    |    | LKB\$B_RQMODE    | = 00000034   |    |
| CHFSL_SIG_NAME  | = 00000004    |    | LKB\$B_STATE     | = 00000036   |    |
| CONVERT         | = 00000000    |    | LKB\$K_LENGTH    | = 00000060   |    |
| CONVSTR         | = 0000016B R  | 03 | LKB\$L_BLKASTADR | = 00000020   |    |
| CONVSTR1        | = 000000C6 R  | 03 | LKB\$L_CSID      | = 00000058   |    |
| CONVSTR2        | = 000000DF R  | 03 | LKB\$L_LKID      | = 00000030   |    |
| CONVSTR3        | = 00000108 R  | 03 | LKB\$L_OWNQFL    | = 00000040   |    |
| CONV_STR_TBL    | = 0000003E R  | 03 | LKB\$L_PARENT    | = 00000048   |    |
| COUNT           | = 00000194 R  | 02 | LKB\$L_PID       | = 0000000C   |    |
| CREAD           | = 000001EC R  | 03 | LKB\$L_REMLKID   | = 00000054   |    |
| CSID            | = 0000023F R  | 02 | LKB\$L_RSB       | = 00000050   |    |
| CV_STRING       | = 00000206 R  | 03 | LKB\$L_SQFL      | = 00000038   |    |
| CWRITE          | = 000001EF R  | 03 | LKB\$V_MSTCPY    | = 00000004   |    |
| DIRENTRY        | = 00000257 R  | 02 | LKB\$W_FLAGS     | = 00000028   |    |
| DIR_ENTRY       | = 000002DB R  | 03 | LKB\$W_REFCNT    | = 0000004C   |    |
| DISPLAY_LOCK    | = 0000051D RG | 03 | LKB\$W_STATUS    | = 0000002A   |    |
| DMP1            | = 000002CF R  | 02 | LKB_BFR          | 00000008 R   | 02 |
| DMP3            | = 000002EF R  | 02 | LKB_RSB_BFR      | 0000006C R   | 02 |
| DMP5            | = 0000030F R  | 02 | LKID             | 0000026F R   | 02 |
| DMP7            | = 0000032F R  | 02 | LKID_BFR         | = 000001A2 R | 02 |
| DONE            | = 00000442 R  | 03 | LKID_OFF         | = 00000000   |    |
| EMODE           | = 000002F7 R  | 03 | LKMODE_TBL       | 000001D1 R   | 03 |
| EX              | = 000001F8 R  | 03 | LOCAL            | 000002BA R   | 03 |
| EXIT            | = 00000F2A R  | 03 | LOCKID           | 00000090 RG  | 02 |
| FAO_GROUP_DSC   | = 000001AE R  | 02 | LOCKMODE_TBL     | 000001C5 R   | 03 |
| FAO_RMINFO_DSC  | = 00000094 R  | 02 | LOCKSTR1         | 0000005A R   | 03 |
| FLAGS1          | = 00000277 R  | 02 | LOCKSTR2         | 0000006F R   | 03 |
| FLAGS2          | = 00000293 R  | 02 | LOCKSTR3         | 00000093 R   | 03 |
| FLAGS3          | = 000002AF R  | 02 | LOCK_COND_HAND   | 000009E8 RG  | 03 |
| FLAGS_TBL       | = 0000022A R  | 03 | LOCK_COUNT       | 00000000 R   | 02 |
| FORMAT_LOCK     | = 00000556 RG | 03 | LOCK_HEAD        | 00000000 R   | 03 |
| FORMAT_RSB      | = 00000BCC RG | 03 | LOCK_STR_TBL     | 0000002E R   | 03 |
| GETLCK          | = 000003C7 R  | 03 | MASTER           | 000002C8 R   | 03 |
| GET_LKB         | = 00000443 RG | 03 | MODE1            | 0000028F R   | 02 |
| GGMOD           | = 000001EF R  | 02 | MODE2            | 000002AB R   | 02 |
| GRANT           | = FFFFFFFF    |    | MSG\$_LOCKIDZER  | *****        | 03 |
| GRANTSTR        | = 00000141 R  | 03 | MSG\$_NOLKB      | ***** X      | 03 |
| GRMD_BFR        | = 0000019C R  | 02 | MSG\$_NOLOCKS    | ***** XX     | 03 |
| GRMD_OFF        | = 00000005    |    | MSG\$_NOPRLOCK   | ***** XX     | 03 |
| GROUP_BFR       | = 0000006C R  | 02 | MSG\$_NORESOURC  | ***** XX     | 03 |
| GROUP_BU        | = 000001BA R  | 02 | MSG\$_OUTOFRANG  | ***** XX     | 03 |
| GROUP_BU_DSC    | = 000001C0 R  | 02 | MSG\$_SUCCESS    | ***** XX     | 03 |
| GROUP_DSC       | = 000001D8 R  | 02 | NEW PAGE         | ***** XX     | 03 |
| GROUP_NUM       | = 000001E8 R  | 02 | NLOCK            | 00000430 R   | 03 |

|               |            |    |    |                  |             |          |
|---------------|------------|----|----|------------------|-------------|----------|
| NOT_VALID     |            |    |    |                  |             |          |
| NSPACE        | 0000031F   | R  | 02 | RSB\$B_RMOD      | =           | 0000004E |
| NULL          | 000001E9   | R  | 03 | RSB\$B_RSNLEN    | =           | 0000004F |
| NULL_CSTRING  | 00000222   | R  | 03 | RSB\$K_LENGTH    | =           | 00000050 |
| PAGE_SIZE     | *****      | X  | 03 | RSB\$K_MAXLEN    | =           | 0000001F |
| PARID         | 00000283   | R  | 02 | RSB\$L_CSID      | =           | 00000038 |
| PARID_BFR     | 00000004   | R  | 02 | RSB\$L_CVTQFL    | =           | 00000018 |
| PART2         | 0000067E   | R  | 03 | RSB\$L_GRQFL     | =           | 00000010 |
| PCBSL_LOCKQFL | = 00000104 |    |    | RSB\$L_PARENT    | =           | 00000048 |
| PID           | 00000273   | R  | 02 | RSB\$L_VALSEQNUM | =           | 0000003C |
| PREAD         | 000001F2   | R  | 03 | RSB\$L_WTQFL     | =           | 00000020 |
| PRINT         | *****      | X  | 03 | RSB\$Q_VALBLK    | =           | 00000028 |
| PRINT_LINE    | 00000F2B   | RG | 03 | RSB\$T_RESNAM    | =           | 00000050 |
| PRINT_LOCK    | 00000856   | RG | 03 | RSB\$V_DIRENTRY  | =           | 00000000 |
| PRINT_RSB     | 00000D34   | RG | 03 | RSB\$W_VALINVLD  | =           | 00000001 |
| PROCESS       | 000002C0   | R  | 03 | RSB\$W_BLKASTCNT | =           | 00000042 |
| PROCESS_QUEUE | 00000E55   | RG | 03 | RSB\$W_GROUP     | =           | 0000004C |
| PRSB          | 000001F3   | R  | 02 | RSB\$W_REFCNT    | =           | 00000040 |
| PWRITE        | 000001F5   | R  | 03 | RSBCSID_BFR      | =           | 0000000E |
| QUEUE_COUNT   | 00000119   | R  | 02 | RSB_BFR          | 00000068 R  | 02       |
| QUEUE_TYPE    | 00000198   | R  | 02 | RSNCEN_BFR       | 0000011D R  | 02       |
| QUE_STR_TBL   | 0000004E   | R  | 03 | SAVE_LOCK_DATA   | 0000006F R  | 02       |
| RACMOD        | 00000243   | R  | 02 | SEQNOM           | 000004B7 RG | 03       |
| REQMEM        | *****      | X  | 03 | SET_HEADING      | 00000213 R  | 02       |
| RES1          | 000002C7   | R  | 02 | SHOW_ALL_LOCKS   | ***** X     | 03       |
| RES2          | 000002C3   | R  | 02 | SHOW_ALL_RES     | 00000328 RG | 03       |
| RES3          | 000002E7   | R  | 02 | SHOW_ONE_LOCK    | 00000A32 RG | 03       |
| RES4          | 000002E3   | R  | 02 | SHOW_ONE_RES     | 0000036E RG | 03       |
| RES5          | 00000307   | R  | 02 | SHOW_PROC_LOCK   | 00000AF3 RG | 03       |
| RES6          | 00000303   | R  | 02 | SHOW_QUEUES      | 000003AE RG | 03       |
| RES7          | 00000327   | R  | 02 | SHOW_RSB         | 00000B92 RG | 03       |
| RES8          | 00000323   | R  | 02 | SKIP_LINES       | 00000B62 RG | 03       |
| RESN1         | 0000021F   | R  | 02 | SMODE            | ***** X     | 03       |
| RESN2         | 0000021B   | R  | 02 | SPACE            | 00000302 R  | 03       |
| RESN3         | 00000233   | R  | 02 | SRSBCT           | 000002CF R  | 03       |
| RESN4         | 0000022F   | R  | 02 | SSS_RESIGNAL     | ***** X     | 03       |
| RESN5         | 0000024B   | R  | 02 | SSS_UNWIND       | ***** X     | 03       |
| RESN6         | 00000247   | R  | 02 | STATE1           | 00000287 R  | 02       |
| RESN7         | 00000263   | R  | 02 | STATE2           | 000002A3 R  | 02       |
| RESN8         | 0000025F   | R  | 02 | STATUS_TBL       | 00000272 R  | 03       |
| RESNAM_BFR    | 00000070   | R  | 02 | STS1             | 000002D3 R  | 02       |
| RESNLEN       | 0000022B   | R  | 02 | STS2             | 000002F3 R  | 02       |
| RES_COUNT     | 00000115   | R  | 02 | STS3             | 00000313 R  | 02       |
| RES_HEAD      | 00000015   | R  | 03 | SUBLKS           | 0000029F R  | 02       |
| RLEN          | 000002DF   | R  | 02 | SYSSFAO          | ***** X     | 03       |
| RMINFO        | 00000337   | R  | 02 | SYSSUNWIND       | ***** X     | 03       |
| RMINFO_BFR    | 000000C5   | R  | 02 | SYS_DSC          | 000001C8 R  | 02       |
| RMINFO_CNT    | 000000C4   | R  | 02 | TYPE             | 00000333 R  | 02       |
| RMINFO_DSC    | 000000BC   | R  | 02 | UMODE            | 0000030E R  | 03       |
| RMINFO_LEN    | 000000B8   | R  | 02 | VAL1             | 0000020F R  | 02       |
| RMOD_BFR      | 0000006E   | R  | 02 | VAL2             | 0000020B R  | 02       |
| RNSPACE       | 0000025B   | R  | 02 | VAL3             | 00000207 R  | 02       |
| RQMD_BFR      | 0000019F   | R  | 02 | VAL4             | 00000203 R  | 02       |
| RQMD_OFF      | = 00000004 |    |    | VALID            | 00000217 R  | 02       |
| RSB           | = 000001EB | R  | 02 | WAIT             | = 00000001  |          |
| RSB\$B_CGMODE | = 0000000D |    |    | WAITSTR          | 0000019B R  | 03       |
| RSB\$B_GGMODE | = 0000000C |    |    | WT_STRING        | 00000214 R  | 03       |

```
+-----+
! Psect synopsis !
+-----+
```

## PSECT name

|          | Allocation        | PSECT No. | Attributes  |
|----------|-------------------|-----------|---|
| ABS .    | 00000000 ( 0.)    | 00 ( 0.)  | NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE |
| \$ABSS   | 00000000 ( 0.)    | 01 ( 1.)  | NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE       |
| SDADATA  | 0000033B ( 827.)  | 02 ( 2.)  | NOPIC USR CON REL LCL NOSHR NOEXE RD WRT NOVEC LONG     |
| LOCK     | 00000F92 ( 3986.) | 03 ( 3.)  | NOPIC USR CON REL LCL NOSHR EXE RD NOWRT NOVEC LONG     |
| LITERALS | 0000034C ( 844.)  | 04 ( 4.)  | NOPIC USR CON REL LCL NOSHR EXE RD NOWRT NOVEC BYTE     |

```
+-----+
! Performance indicators !
+-----+
```

## Phase

| Phase                  | Page faults | CPU Time    | Elapsed Time |
|------------------------|-------------|-------------|--------------|
| Initialization         | 29          | 00:00:00.05 | 00:00:01.24  |
| Command processing     | 107         | 00:00:00.45 | 00:00:03.92  |
| Pass 1                 | 310         | 00:00:06.76 | 00:00:26.79  |
| Symbol table sort      | 0           | 00:00:00.69 | 00:00:01.98  |
| Pass 2                 | 264         | 00:00:02.72 | 00:00:10.19  |
| Symbol table output    | 26          | 00:00:00.14 | 00:00:00.30  |
| Psect synopsis output  | 3           | 00:00:00.02 | 00:00:00.02  |
| Cross-reference output | 0           | 00:00:00.00 | 00:00:00.00  |
| Assembler run totals   | 741         | 00:00:10.83 | 00:00:44.44  |

The working set limit was 1650 pages.

66653 bytes (131 pages) of virtual memory were used to buffer the intermediate code.

There were 40 pages of symbol table space allocated to hold 660 non-local and 73 local symbols.

1491 source lines were read in Pass 1, producing 79 object records in Pass 2.

26 pages of virtual memory were used to define 24 macros.

```
+-----+
! Macro library statistics !
+-----+
```

## Macro library name

## Macros defined

| Macro library name                 | Macros defined |
|------------------------------------|----------------|
| \$255\$DUA28:[SDA.OBJ]SDALIB.MLB;1 | 9              |
| \$255\$DUA28:[SYS.OBJ]LIB.MLB;1    | 3              |
| \$255\$DUA28:[SYSLIB]STARLET.MLB;2 | 9              |
| TOTALS (all libraries)             | 21             |

660 GETS were required to define 21 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LI\$\$:LOCK/OBJ=OBJ\$:LOCK MSRC\$:LOCK/UPDATE=(ENH\$:LOCK)+EXECML\$/LIB+LIB\$:SDALIB/LIB

0352 AH-BT13A-SE  
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION  
CONFIDENTIAL AND PROPRIETARY

HANDLER  
LIS

DUMP  
LIS

MAIN  
LIS

MAPPING  
LIS

EXAMPSL  
LIS

MMG  
LIS

INDEX  
LIS

LOCK  
LIS

PARSE  
LIS